Défense nationale National Defence

UNCLASSIFIED

DEFENCE RESEARCH ESTABLISHMENT
CENTRE DE RECHERCHES POUR LA DÉFENSE
VALCARTIER, QUÉBEC

DREV - TM - 9813

Unlimited distribution / Distribution illimitée

INFRARED EYE: MICROSCANNING

by

Jean Fortin and Paul Chevrette

June/juin 1998

RESEARCH AND DEVELOPMENT BRANCH
DEPARTMENT OF NATIONAL DEFENCE
CANADA
BUREAU - RECHERCHE ET DÉVELOPPEMENT
MINISTÈRE DE LA DÉFENSE NATIONALE

Canada

SANS CLASSIFICATION

UNCLASSIFIED

DEFENCE RESEARCH ESTABLISHMENT
CENTRE DE RECHERCHES POUR LA DÉFENSE
VALCARTIER, QUÉBEC

DREV - TM - 9813

Unlimited distribution / Distribution illimitée

INFRARED EYE: MICROSCANNING

by

Jean Fortin and Paul Chevrette

June/juin 1998

Approved by/approuvé par

Section Head/Chef de section

16/06/98
Date

SANS CLASSIFICATION

ABSTRACT

This memorandum describes the research and development work that was undertaken at DREV to build a microscanning system to enhance the resolution of a focal plane array camera. This work was conducted as part of the Infrared Eye project, which shows a new approach that intends to help in the detection and identification problems related to search and rescue operations.

The microscanning system is based on a focal plane array camera and can improve the resolution by a factor of two. It can operate over all the camera frame rate range (240 Hz) with no degradation in the image quality.

It includes an optical component that can perform the microdisplacements of the image on the focal plane array with electronics as well as its associated software allowing synchronization, timing and real-time image processing. This memorandum thoroughly describes the innovative approach used to perform the microscanning operation.

RÉSUMÉ

Ce mémorandum décrit les travaux de recherche et de mise au point d'un système à microbalayage pour l'amélioration de la résolution d'une caméra à matrice au plan focal. Le travail a été effectué au CRDV dans le cadre du projet Oeil infrarouge, qui se veut une nouvelle approche développée en vue de résoudre les problèmes de détection et d'identification reliés aux opérations de recherche et sauvetage.

Le système est basé sur une caméra à matrice au plan focal et permet d'augmenter la résolution des images par un facteur de deux. Il peut fonctionner jusqu'à la cadence maximale de la caméra (240Hz) et ce sans dégradation apparente du résultat.

Il comprend une partie optique destinée à la réalisation de l'opération de microbalayage proprement dite ainsi qu'une partie électronique et logicielle permettant la synchronisation, le contrôle et le traitement des images en temps réel. Le mémorandum décrit en détail l'approche originale utilisée pour effectuer le microbalayage.

TABLE OF CONTENTS

EXECUTIVE SUMMARY

In the recent years, infrared (IR) imaging system has moved to a new sensor technology known as the Focal Plane Array (FPA) technology. This technology uses a two-dimensional matrix of detectors in the image plane of the objective lens, to acquire the IR image of the scene. The FPA thus acts similarly to a photographic film and eliminates all scanning mechanisms as used in earlier versions of IR imaging systems, also known as Forward Looking IR (FLIR) systems. The advantages of the technology are obvious, from the elimination of mechanical complexity with corresponding weight and size reduction to significant improvements in performance and dynamic range because of the capability to adjust various parameters such as frame rate and integration time.

The major draw back of FPAs is a loss in resolution by a factor two as compared with a scanning system with the same optics and size of detector as the individual detectors in the FPA. This loss is due to the fixed sampling of the scene by individual detectors in the array, which is reduced to one sample per detector size as opposed to infinite sampling for scanning systems, at least in the direction of scan. This undersampling introduces aliasing at higher frequencies in the reconstruction of the image. To eliminate that effect and obtain an image frequency content limited only by the detector size, there exists a technique called microscanning which consists in building the final full-resolution IR image by interlacing a sequence of sub-images where the FPA has been displaced by a fraction of its detector size with respect to the image scene, in both dimensions. The result is an over-sampled IR image with the full frequency content allowed by the detector size.

This memorandum describes the research and development work that has been undertaken to develop and build a microscanning camera system. The work was conducted as part of the Infrared Eye project which shows a new concept of an infrared surveillance camera system aimed at improving the effectiveness of search an rescue operations. The system is based on two IR cameras operating simultaneously, one covering a wide field-of-view (WFOV) and the other one having a narrow field-of-view (NFOV) mobile within the WFOV. The WFOV camera is optimized for detection while

the NFOV camera is optimized for resolution. The resolution of that camera is improved using a technique based on microscanning.

The microscanning technique described in this document and implemented in the project differs from others currently used and is the object of a patent request. It consists in imposing micro-displacements to the imaging lens in front of the FPA by means of piezo-electric micro-positioners and special flexible supports for the lens in both X and Y axis. This support method allows for a symmetrical mechanical loading for the two axis.

The advantages of the developed microscanning technique and device are that it can be synchronized to the adjustable frame rate and integration time of the camera, and any microscanning mode and path can be programmed electronically. In our application, we can select 2x2, 3x3 and 4x4 microscanning steps, as opposed to more conventional techniques using a wheel of prisms with fixed speed and mode. Also, the obtained image displacement is equal to the displacement imposed to the lens, irrelevant of its focal length, hence allowing a change of the lens or telescope which may be used as front optics without having to modify the microscanning system.

The implementation of this microscanning technique for the NFOV camera in the IR Eye project significantly improve its resolution and therefore its performance as a component of the system, to the benefit of search and rescue operations or other surveillance applications pertinent to the Canadian Forces, for which it could be used. The advantages of the FPA technology to replace other fielded scanning FLIR systems by the Forces may be shadowed by their loss of resolution which cannot be compensated by a reduction in the size of the detectors, already limited by diffraction and fabrication constraints. This microscanning technique has the potential of doubling their resolution while preserving their flexibility for dynamic range adjustment.

LIST OF ABBREVIATIONS

AE-RTAS	Amber Engineering Real Time Application Software
CPLD	Complex Programmable Logic Device
FLIR	Forward looking infrared
FPA	Focal plane array
HSVB	High speed video bus
IFOV	Instantaneous field of view
LSF	Line spread function
MDTF	Modulation detector transfer function
MTF	Modulation transfer function
NFOV	Narrow field of view
NUC	Non-uniformity correction
WACISS	Wide area coverage infrared search system
WFOV	Wide field of view

1.0 INTRODUCTION

The Infrared Eye exhibits a new concept for an infrared surveillance camera firstly aimed at solving some of the present FLIR system deficiencies and secondly increasing search and detection efficiency. In current FLIR systems, a single detector element used in conjunction with a scanning mechanism is used to build an image of the scene. The user can switch between a wide field of view (WFOV) mode during the search and detection processes and a narrow field of view (NFOV) for recognition and identification. However, because of the increased sensitivity of the NFOV mode over the WFOV mode, the operator generally prefers to keep the system on the NFOV setting and performs a systematic search of the whole area. This eventually leads to user fatigue and an increased search time as well as a lack of situation awareness.

The strategy used in that work to solve that problem is to mimic some of the properties of the human eye. This is nowadays possible by using the new technology offered by staring focal plane arrays (FPA). It is well known that the light sensitive part of the human retina is composed of two kinds of cells, the rods and the cones. The cones are mainly located in the center at a point called the fovea while the rods are located around. It is also known that the cones are less sensitive and more tightly packed than the rods. For these reasons, the central part of the eye sees with high resolution and low sensitivity while the peripheral part sees with low resolution and high sensitivity.

The Infrared Eye will mimic these two properties of the human eye by fusing information obtained from two infrared FPA cameras onto a single display. The two cameras will respectively image a NFOV and a WFOV. Microscanning will be used to increase the image resolution in the NFOV and an optimization of the acquisition process in conjunction with image processing will be used to increase the sensitivity of WFOV images.

This memorandum is the second of a series on the development of the Infrared Eye and is concerned with the microscanning problem. Firstly, the microscanning operation is introduced. Secondly, the microscanning system developed is presented and finally, results are given. The memorandum concludes on some remarks about future phases of the project.

The work covered by this document was performed at DREV between September 1992 and July 97 under Task DASP-160, "Wide Area Coverage Infrared Search Systems (WACISS)" and under Thrust 3D, Work Unit 3DA11 (now continuing under 3DA14, IR Eye Second Generation).

2.0 THE MICROSCANNING PROCESS

Infrared FPAs are becoming widely used in the field of thermal imaging. As compared to older single detector scanning systems, those sensors show tremendous improvements. Among them, the opportunity to adjust the integration time according to the scene radiation and the great reliability of integrated systems are just some examples. FPAs however have a major drawback over scanning systems: for a given instantaneous field of view (IFOV), they reproduce only half the frequencies that can be represented by scanning systems. This is due to the Nyquist theorem which stipulates that at least two samples must be taken for each frequency cycle (Ref. 1) in the image.

As part of the Infrared Eye project, it was decided to investigate on the microscanning process to determine if the resolution of a staring array imager can be increased at the cost of some extra mechanical parts and image processing. As we found, microscanning was successful at eliminating the loss in resolution due to discrete sampling of FPA while it maintained most of the benefits of the technology.

The microscanning technique can be seen as an oversampling process (Refs. 2 and 3). A series of images representing the same scene is captured while each time displacing the image over the array by a fraction of the detector pitch. A microscanned image can then be built by interlacing all the pixels of the images in a checkerboard like pattern. For the remaining of the text, a low-resolution image will be called a sub-image while an image constructed from the microscanning process will be called an oversampled image.

The Fig. 1 gives a representation of the microscanning operation. On the left hand side of Fig. 1, four sub-images obtained from a 2x2 microscanning cycle i.e. two horizontal and two vertical steps are shown. Each step corresponds to a movement of the detector matrix equal to half of the detector pitch. Referring to Fig. 1, a first image is acquired with the matrix in position one. Then, the matrix is moved one step to the right and a second image is acquired. Next, the matrix is moved down one step and the third image is acquired and finally, the cycle is closed by moving the array one step to the left and acquiring the fourth image. The right hand side of Fig. 1 represents the resulting oversampled image obtained by interlacing all the pixels of the previous sub-images.

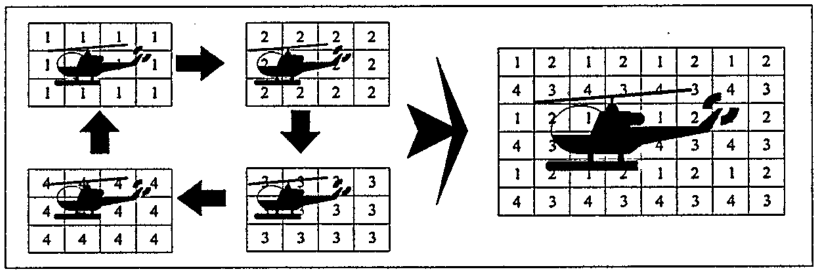


FIGURE 1 - Representation of the microscanning process

The microscanning operation is not limited to a four step cycle. It can be executed in any number of steps or can be restrained to displacements along a single direction e.g. horizontally or vertically. As will be shown in the remaining of this chapter, a compromise must be done between the final oversampled image frame rate and the resolution improvement.

The best way to characterize the utility of microscanning is firstly to compare the resolution of a single detector scanning system to the resolution of a staring array imaging system and secondly to study the resolution improvement due to microscanning. Such a comparison will be discussed in the following three sections.

2.1 Resolution of Single Detector Scanning Systems

The maximum resolution of a single detector scanning system can be obtained from the modulation transfer function (MTF) (Ref. 4). The first zero in the MTF curve gives the maximal frequency that a single detector system can reproduce.

To determine the MTF one has to calculate the modulus of the Fourier transform of the impulse response of the imaging system. For the sake of simplicity, the measurement is generally done using a vertical slit subtending an angle much smaller than the instantaneous field of view (IFOV) of the camera and the impulse response is obtained from a line of the video signal passing through the image of the slit.

Analytically, the MTF can be represented by Eq. 1 where $\Im$ and $\otimes$ stand respectively for the Fourier transform and the convolution operation, $\delta(u)$ represents an impulse (Dirac function) and $h(u)$, the detector transfer function expressed in the spatial domain (u).

$$MTF = \left|\Im\left[\delta(u) \otimes h(u)\right]\right| \tag{1}$$

In an ideal system *h(u)* would be represented by a rectangular function of width equal to the detector size. If the detector size is normalized with respect to the focal length of the lens used for the measurement, then the result can be expressed in terms of cycle per unit of angle (mrad).

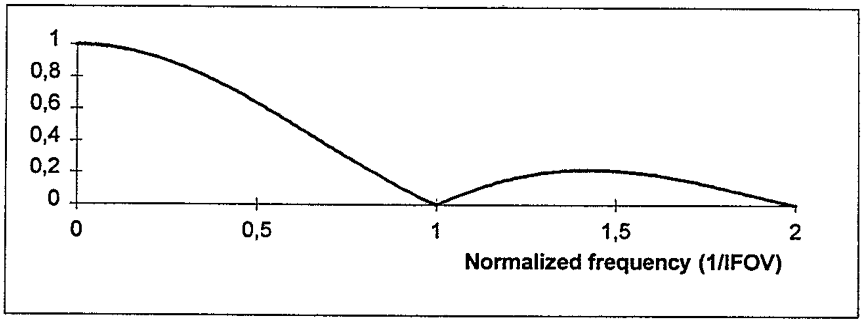


FIGURE 2 - MTF of a single detector scanning system

Fig. 2 shows the MTF of a single detector imaging system. The first zero of the function indicates that the maximal resolved frequency is given by one cycle in the IFOV.

2.2 Resolution of Staring Array Imaging Systems

In the case of staring array imaging systems, the maximum resolvable frequency is given by the sampling theorem due to the discrete nature of the process. The MTF is no longer defined and the modulation detector transfer function (MDTF) (Refs. 5 and 6) was introduced instead as a means to characterize FPAs. In theory, the MDTF is defined by the modulus of the Fourier transform of the impulse response of a single detector. In practice, if one assumes that all detectors are uniform, then the horizontal MDTF can be measured by showing the imaging system with a narrow vertical slit smaller than a single detector IFOV. Then, the MDTF is obtained by computing the modulus of the Fourier transform of the signal produced by a line of detectors passing through the image of the slit. It is possible to determine the maximum resolvable frequency of a staring array imaging system by localizing the first zero of the MDTF function.

Two cases of interest are shown on Fig. 3. Firstly, case A where the signal impulse falls on a single detector and secondly, case B where the signal impulse falls just in between two detectors. In that case, the energy is divided equally and stimulates two adjacent detectors leading to a wider impulse response for the matrix.

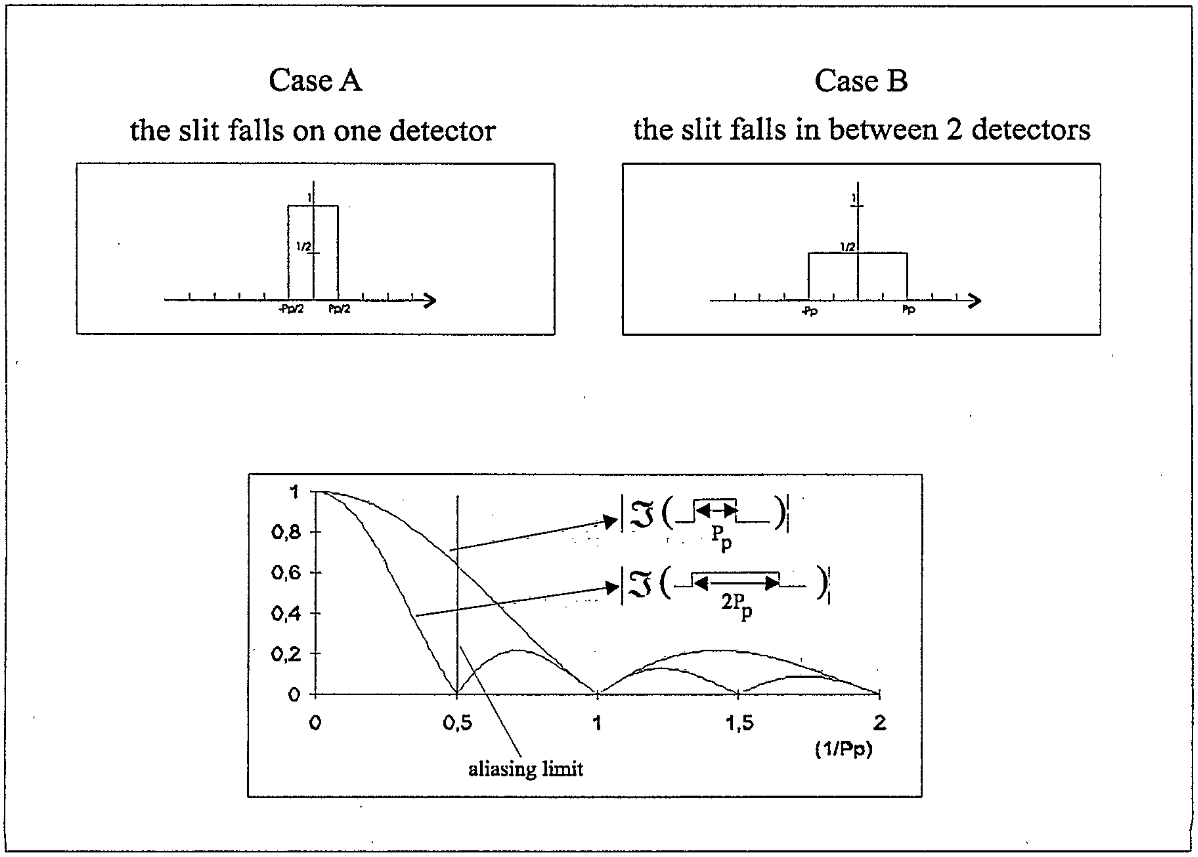


FIGURE 3 - Graphical representation of the MDTF

The bottom part of Fig. 3 shows the MDTF curve for both cases. Case A is referred to as the best MDTF while case B is named the worst MDTF curve. The performance of a matrix is generally expressed in terms of the worst MDTF curve as it corresponds to the Nyquist theorem that stipulates that one cycle must be sampled by at least two detectors.

By comparing the results of Fig. 2 and 3, it is seen that for the same detector size and optics, a staring array system will only reproduce half the frequencies a scanning system does. In that respect, one-detector scanning systems are more efficient than FPAs and new methods must be used to recover the information lost and get the best of each detector. The microscanning technique seems to be the best approach to solve that problem.

2.3 Resolution of Microscanned Images

The resolution of microscanned images can be studied using a modified MDTF. The measurement procedure is about the same except that the output signal of the array must be processed according to the microscanning pattern chosen. Fig. 4 shows the MDTF measurement for a linear array of four detectors and a two-step microscanning operation.

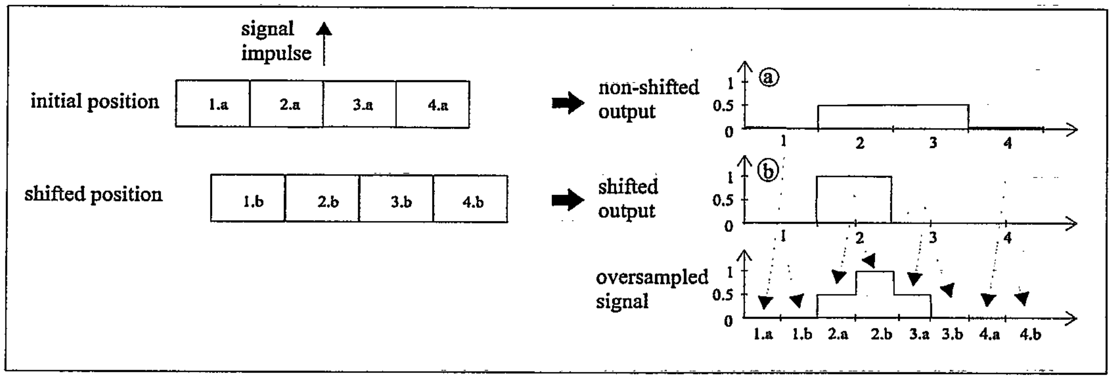


FIGURE 4 - MDTF measurement for a linear array of four detectors and a two-step microscanning operation

The left hand part of Fig. 4 shows a signal impulse falling on the detector array in an initial position (a) and a shifted position (b). To characterize the worst case, the impulse was localized in between detector two and three on the array in its initial position. The right hand part of Fig. 4 shows the output of each array and the lower graphic shows the oversampled signal obtained following microscanning data processing. The modified MDTF is obtained by computing the modulus of the Fourier transform of this oversampled signal.

Fig. 5 shows the MDTF for a two-, three- and four-step microscan operation. The horizontal axis on Fig. 5 has been normalized with respect to the detector pitch. The dashed curve on Fig. 5 was added for comparison. It shows the MDTF of a non-microscanning array having the same IFOV.

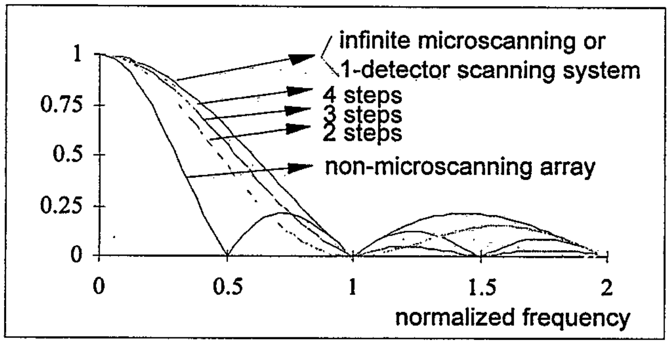


FIGURE 5 - MDTF of microscanned images

As seen on Fig. 5, a two-step microscan operation brings the first zero of the MDTF to the same level it is with a one-detector scanning system having the same IFOV. However, the contrast of high frequencies is slightly attenuated. Increasing the number of microscanning steps reduces this artifact. For an infinite number of steps, the shape of the curve is exactly the same as the one obtained with a one-detector scanning system. Thus, the gain obtained by increasing the number of microscanning steps decreases gradually. A compromise must be reached between the required oversampled image frame rate, the amount of data to process and the error in the modified MDTF curve.

The effect of the detector size versus the detector pitch can be studied in a similar way. In that case, the first zero of the MDTF for a microscanned array appears at one over the detector size instead of the detector pitch meaning that even higher frequencies can be resolved by the system.

2.4 Microscanning Methods

In order to implement the image motion over the FPA, some sort of moving mechanism is required. Some methods are described in the literature, each of them being characterized by advantages and drawbacks. The most straightforward method would be to move the array itself. This method is rarely retained as a good solution because the array is generally enclosed in the heavier piece of the imaging system. Precise and rapid displacement would require a very sophisticated and complex system. A more suitable method would require the use of optical prisms (Ref. 7). A schematic view of that method is shown on Fig. 6.

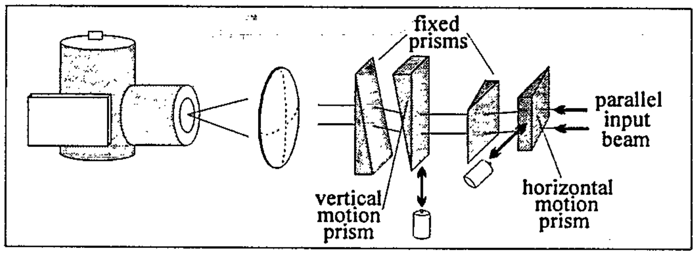


FIGURE 6 - Prism based microscanning method

The propagation of light through the optical system can be visualized by considering that a parallel beam enters the optics. The beam is first deviated by the horizontal motion prism and brought back on its way (with some shifting) by a fixed prism. The displacement is proportional to the space left between the two prisms which can be adjusted using any kind of micro-positioning device. The vertical motion is realized using a similar arrangement of prisms. Finally, the image of the scene is formed onto the FPA with a focusing lens mounted in between the microscanning arrangement and the detector matrix.

This configuration has the advantage of being very light in weight and requires that a single optical element has to be moved on each motion axis. Moreover, it can be optimized and reduced to just one pair of prisms.

A variation of this method is used by FLIR System Inc. in a commercial version of microscanning infrared imager. The micro-scanner is based on a flat transparent window of a certain thickness. To realize the required displacements, the window is tilted by an angle proportional to the deviation required. The deviation induced can be calculated as a function of the index of refraction at the wavelength of the light used, the thickness of the window and the tilt angle.

The main drawback of the prism-based microscanning device is related to the chromatic aberration. The amount of chromatic aberration depends upon the distance between the prisms and varies for each microscan position. This might be difficult to correct for and may be intolerable in a system covering wide waveband.

Another method of doing microscanning is to use tilting mirrors (Ref. 8) as shown on Fig. 7. The use of reflective surfaces eliminates the problem of chromatic aberrations.

However, it requires folding of the optical path which might be cumbersome in some situations. In the case of two-dimensional motion, the use of mirrors introduces a rotation in the image plane. In some cases, this is not negligible and may be difficult to compensate for.

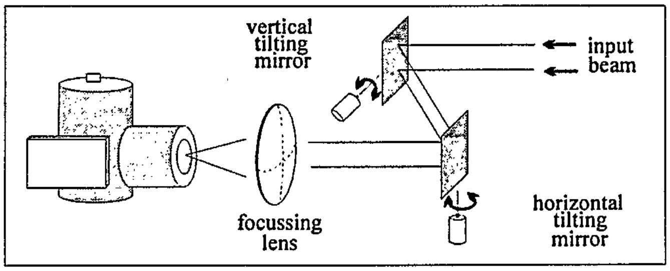


FIGURE 7 - Mirror based microscanning method

A non-mechanical microscanning method was recently presented by Barnard et al. (Ref. 9). It is based on a liquid-crystal beam steering device mounted in front of the detector matrix as shown on Fig. 8. The beam steering device operates as a prism deviating the input beam.

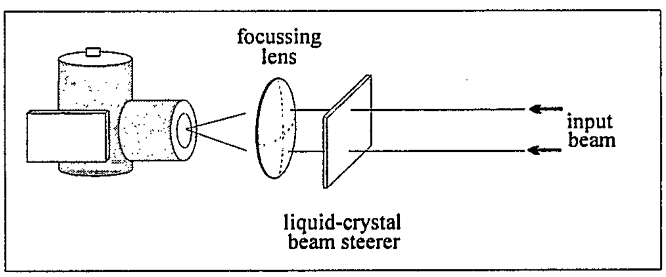


FIGURE 8 - Liquid-crystal based microscanning method

The method seems attractive at first glance. However it is limited by two important factors. Firstly, the transmission factor of the liquid-crystal device is low and secondly, the beam steering device must be preceded by a polarizer to operate correctly. This contributes to further reduce the transmission factor. Barnard et al. specifies an overall transmission factor for the microscanning device to be less than 12% in the 3 to 3.5μm band.

The best method found to produce a compact and efficient microscanning system is to move the focusing lens placed in front of the detector matrix (Ref. 10). This method has several advantages over the previous ones in terms of aberrations, transmission, symmetry and weight and is easy to implement. Moreover, the impact on the optical quality of the imaging system is negligible as the lens is displaced only by small amounts (e.g. one pixel size (38um)). A schematic diagram of the method is shown on Fig. 9.

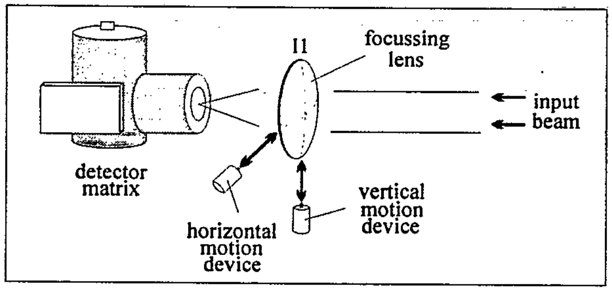


FIGURE 9 - Lens based microscanning method

A parallel beam entering the lens is focused onto the detector matrix. It is possible to show that any displacement of that lens produces an equivalent displacement of the image over the detector array. This is independent of the focal length of the lens used and a two-dimensional movement can be obtained by moving the lens on two-axis. This however requires the use of a two-axis translation table and two perpendicularly mounted micro-translators.

The use of a standard two-axis translation table is generally inappropriate because in that configuration, one axis must support the other leading to excessive and uneven loads on each axis. To achieve high speed performance, the natural frequency of the system must be increased by minimizing the total load and finding a way to split the load equally on each axis. Two methods were studied: firstly, a two-lenses based microscanning system and secondly a special two-axis micro-translation device.

2.4.1 Two-lenses Based Microscanning System

A major improvement over a system based on a standard two-axis translation table can be obtained if the movement is decomposed and realized using separate components: one for the horizontal movements and another one for the vertical movements. Fig. 10

shows a schematic drawing of a two-lenses microscanning system. Lens L_1 is used to control the displacements on one axis while lens L_2 is responsible for displacements on the other axis. When the effect of the two lenses is combined then a two-dimensional motion is obtained.

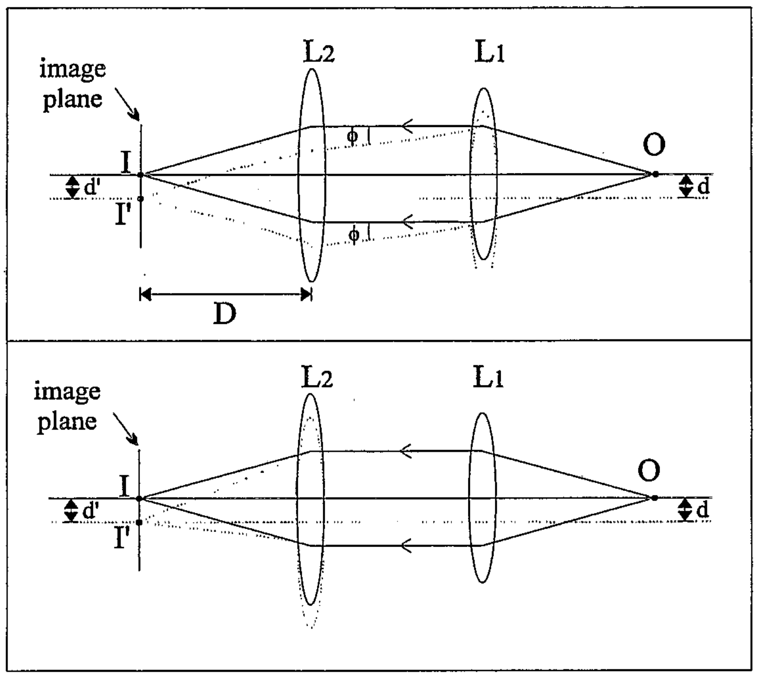


FIGURE 10 - Two-lenses based microscanning method

Referring to the upper part of Fig. 10, rays emerging from the point O cross the lens L_1 and are sent as a parallel beam toward the lens L_2. The latter focuses the beam onto a point I situated on the image plane. If L_1 is moved down by a distance (d), then the rays emerging from the point O come out from L_1 with an angle (ϕ). This angle is related to the focal length of L_1. Then, L_2 focuses the beam at I'. The distance between L_2 and the image plane (D) depends on the focal length of L_2. This implies that the distance (d') between I and I' on the image plane is related to the ratio of the focal length of the two lenses. The lower part of Fig. 10 is essentially the same as Fig. 9. The parallel beam coming form lens L_1 is focused onto point I or I' according to the position of L2.

This concept allows for the construction of a very compact system with equal constraints on both axis. The drawback is that it requires the building of a non conventional lens and the alignment of two optical elements. As another approach to improve the performances of a microscanning system, we have developed a special two-axis micro-translation device.

2.4.2 Single-lens Based Microscanning Method Based on a Two-axis Micro-translation Device

The microscanning operation can be realized using a single lens if the small amplitude of the movement required to realize the microscan steps is taken into account. A two-axis micro-translation device can be built using two flexible push rods mounted on a lens assembly in the configuration shown on FIGURE 11a. The rods must be chosen to be axially stiff but flexible when submitted to a perpendicular force.

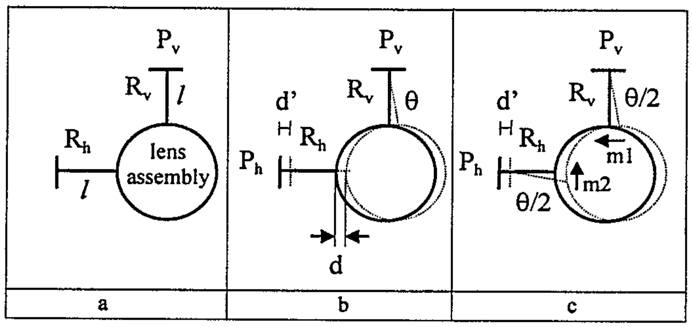


FIGURE 11 - Schematic diagram of the two-axis micro-translation table

In Fig. 11.a, R_h and R_v represent two flexible rods of length *l*. They are tied at one end to the lens assembly and at the other end to piezoelectric translators (P_h and P_v) which allow motion through expansion and contraction. Those are mounted on a rigid frame (not shown on Fig. 11) to allow for easy alignment of the apparatus in front of the detector matrix.

Fig. 11.b and 11.c give an overview of the action and reaction forces occurring following an expansion of the horizontal translator by a length d. This expansion pushes onto the lens assembly and bends the vertical rod (R_v) by an angle (θ). This, in turn, creates a moment (m_1) that rotates the lens assembly counter clockwise as shown on Fig. 11.c. The rotation does not affect the image position as it occurs with respect to the optical axis. It continues until both rods bend equally ($\theta/2$) and develop inverse moments that sum up to zero. The movement of the lens assembly can be established from Fig.

11.c and simple geometry. Eq. 1 gives an analytical representation of effective displacement (d) as compared to the translator elongation (d').

$$d = d' - l(1 - \cos(\theta/2)) \tag{1}$$

Knowing that θ is small, it is possible to neglect the second term of Eq. 1 and find:

$$d \approx d'$$

2.5 Microscanning Advantages and Drawbacks

The process of microscanning presents many advantages and a few drawbacks. The first advantage is that it increases the resolution limit of a given FPA. A factor of two in resolution can be obtained by using a two-step microscan operation. As previously shown, increasing the number of step does not further improve the resolution but enhances the contrast of highest frequency components. The resolution limit could be improved at the cost of a more intense image processing algorithm. This aspect is covered by the super-resolution technique (Refs. 11 and 12) which has not been considered here as it generally requires too much processing time and hence is not applicable in real time situations.

Microscanning also plays a major role in eliminating the effect due to a limited fill factor of the FPA. It produces an overlap in the field of view of adjacent detector elements and reduces the possibility of missing a target image that could fall in between two detectors.

The main drawback of microscanning is that it reduces the overall frame rate. As previously seen, the technique requires the acquisition of a number of sub-images to form an oversampled one (four in the case of a 2x2 operation and up to 16 in the 4x4 mode). Typically, the frame rate of a 256x256 FPA is limited around 240 Hz leading to a maximum oversampled image frame rate of 15 Hz in the 4x4 mode.

Sensitivity is also a major issue in a microscan-based system. It reduces the amount of time available to integrate the signal if a high frame rate is required. This factor can be controlled by choosing an appropriate detector material. InSb detectors are very sensitive and operates normally with short integration times (<3 msec) when used for imaging around room temperature. By choosing such an FPA, a good sensitivity can be maintained even when operated at higher frame rates (240 Hz). Moreover, sufficient time still exist in that case between two integration periods (≈1.17 msec) to displace the image from one microscanning position to the next.

An important requirement to reduce motion blur in microscanning images is that the image must be stabilized over the FPA during integration. Two kinds of FPAs are available: snap shot and rolling ones. Snap shot FPAs operate like standard film camera meaning that all detectors integrate the signal in parallel. Rolling integration FPAs are more complicated. The integration time of each line of the array is delayed by the time required to read out the preceding line. A timing diagram for a rolling integration FPA set with a 50% integration time ratio is shown on Fig. 12. As seen, the readout time of line #1 follows its integration period. The whole timing for line #2 is delayed such as readout occurs at the end of line #1 readout process. The same thing happens for each line of the FPA leading to an overlap of the integration period of successive images. For maximum integration time, the maximum frame integration period spans over as much as two image periods as depicted in Fig. 12.

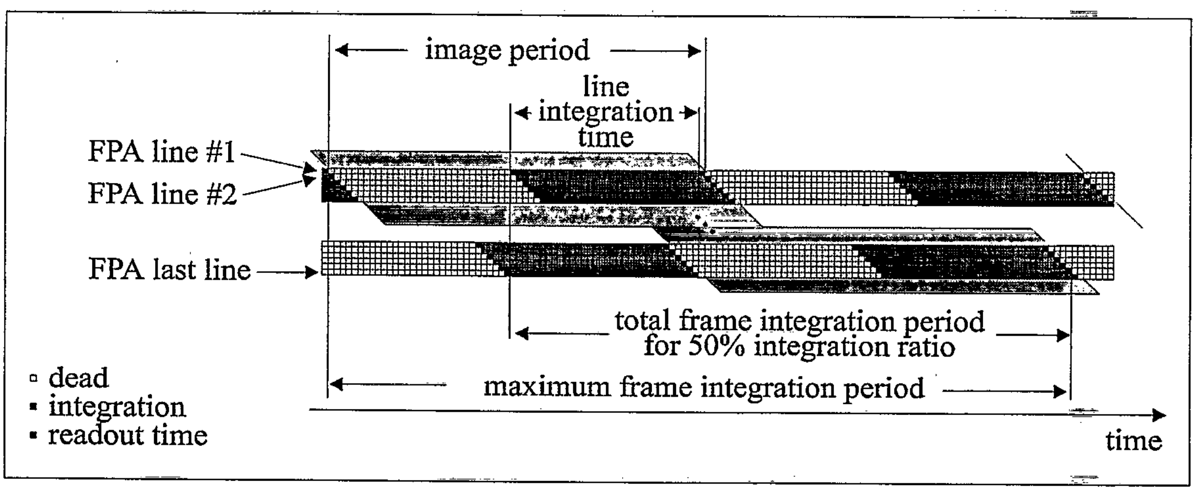


FIGURE 12 - Timing diagram of a rolling integration FPA

When using a rolling integration FPA in a microscanning system, then one must take into account the overlapping in the integration period of successive images. One way to correct for this, is to drop one image over two resulting in a lower final frame rate. This aspect will be covered in detail in the next section.

3.0 THE MICROSCANNING SYSTEM

To improve the resolution of a staring FPA, a microscanning system was developed. The main characteristic of the prototype is that it can operate in real time at a frame rate of up to 240Hz. A schematic diagram of the prototype is shown on Fig. 13. It is composed of three main parts: an imaging system, a microscanning unit and a video unit.

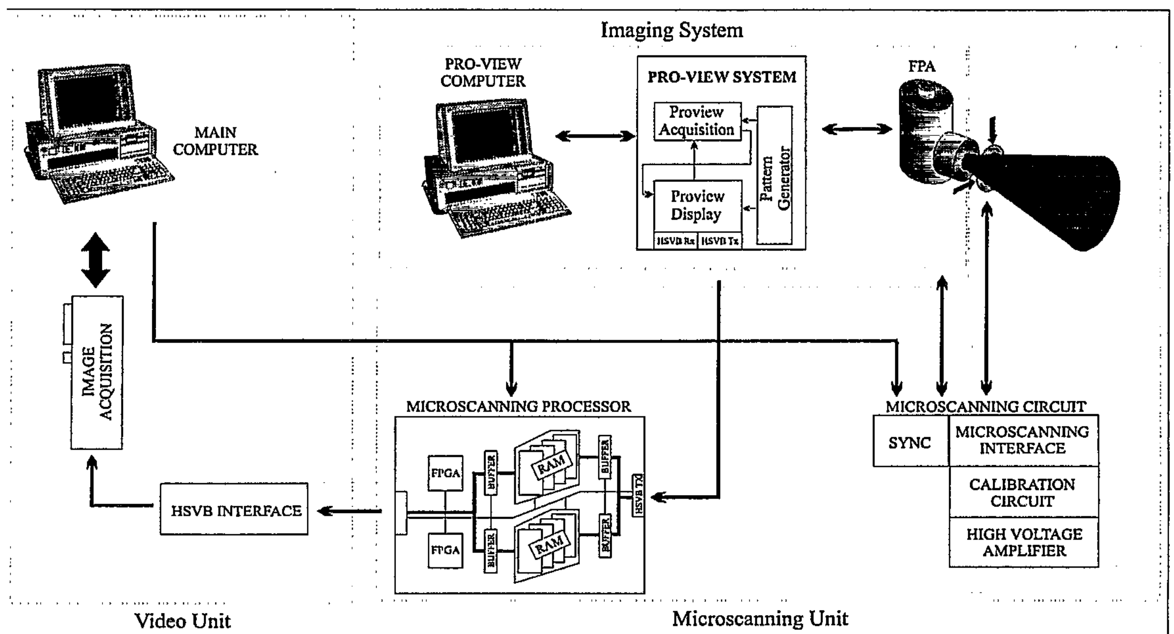


FIGURE 13 - Schematic diagram of the microscanning system

The following sections gives a thorough description of each component.

3.1 The Imaging System

The imaging system includes a camera head and a camera controller. The camera head is made of a detector array, a cooling system and a transimpedance amplifier which relays the analog output signals to the camera controller. This unit allows to set the operating parameters of the FPA (integration time, frame rate, etc.) through software control and to construct the images.

3.1.1 The Camera Head

The heart of the microscanning system is based on a 256x256 InSb FPA manufactured by Amber Engineering (AE4256). It is sensitive in the 3-to-5-μm region. Each detector has a square shape and measures 31 μm. The detector pitch is specified to be 38 μm and a cold shield limits the aperture to f/3. The frame rate is adjustable and can range from 0 to 240 Hz. The FPA operates in the rolling integration mode and the integration time can be varied from about 0 to 99% of the total frame period. A thorough evaluation of this camera system was previously performed and results were published in a previous report (Ref. 13).

3.1.2 The Camera Controller

The camera controller is a Pro-View system developed by Amber Engineering. It includes all of the hardware and software components necessary to build the images produced with the Amber FPA. A thorough description of the system is available in the Pro-View hardware manual (Ref. 14).

Briefly, a Pro-View system includes three electronics boards (an acquisition board, a pattern generator board and a display board) designed to drive one FPA. Up to four Pro-View systems can be mounted in the same enclosure allowing for the independent control of up to four FPAs.

The acquisition board is used to grab the video signal either under an analog or a digital format. It also applies a correction for pixel-to-pixel non-uniformity (NUC). All the timing and clocking signals required to operate this module are produced by the pattern generator board. After the NUC corrections, the samples are sent to the display board which performs bad pixel replacement by means of a vector table and sends the images to the output either under an analog format through standard BNC connectors or under a digital format via a high speed video bus interface (HSVB). By default, the display board is programmed to produce an analog RS-170 compatible signal. Clocking for the display board is generally supplied by an internal oscillator. However, numerous external synchronization sources can be used including the pattern generator. In that case, exact synchronization of the input and output processes can be obtained.

The Pro-View architecture is based on a dual frame buffer approach. While one buffer is used to memorize the input samples, the other one is used to read out a corrected image and send it to the display. This means that the output image is always delayed by one frame period. At the end of a frame period, a swap occurs between the read and write buffers and the process goes on. Priorities can be assigned to determine the swap time in systems operating with different read and write clocks.

The general operation of the Pro-View system is controlled via a software called AE-RTAS. A thorough description of the software is available in the Real time application system user's guide (Ref. 15). The following section gives the proper sequence required to set the Pro-View system in the correct microscanning mode.

3.1.3 Setting up the Camera Controller for Microscanning

To realize microscanning, the Pro-View system must be configured to acquire data from an analog source (the FPA), and to output digital images on the HSVB. The operation of the acquisition and the display board must be synchronized and the frame buffer swapping process must be synchronized with the FPA frame rate in order to make sure that each image is correctly sent to further stages of the microscanning system.

The following section gives the necessary entries to AE-RTAS to properly set the Pro-View system for microscanning. The procedure given considers that the AE-RTAS software is started in the default mode by invoking it with the default configuration file i.e. ae256@4t.aec.

Firstly, the Pro-View system must acquire data from an analog source and to output digital images on the HSVB. This is done by entering the following sequence:

- Main Menu
- Unit Config
  - Mode Config
    - Operation Mode
      - Prebuffer with FPA Acquisition

Secondly, the frame buffer swapping priority mechanism must be set as to produce a swap in conjunction with each FPA vertical synchronization pulse. This is done by entering the following sequence:

Main Menu
Unit Config
Mode Config
Buffer Swap
Fpa Sync

Thirdly, the clock source for the HSVB must tie to the pattern generator clock. One way to do this is to use the single-ended backplane clock input available on the display board and hardwires it to the main clock output of the pattern generator. Then, the following software selection must be done:

Main Menu
HSVB Timing
Clock Source
Single-Ended Backplane Clock

Fourth, the HSVB frame width and height must be adjusted to the FPA dimension. This is done by entering the following sequence:

Main Menu
HSVB Timing
Frame Width: 256
Frame Height: 256

3.2 The Microscanning Unit

The microscanning unit was entirely developed at DREV. It includes an electrooptic part used to realize the microdisplacements of the image over the FPA and all the electronics required to generate the displacements and synchronize them with the FPA frame rate and integration time. It also includes a real time microscanning processor used to build the oversampled images from a sequence of low resolution images. Fig. 13 gives a schematic view of the microscanning unit showing the main communication paths shared with the other constituents of the system. The remaining of this section gives a thorough description of each part.

3.2.1 The Microscanning Head

The microscanning head is the key part of the unit. It allows two-dimensional displacement of the image over the FPA surface. The strategy used is based on the two-axis micro-translation table introduced in section 2.4.2. A photograph of the experimental setup is shown on Fig. 14.

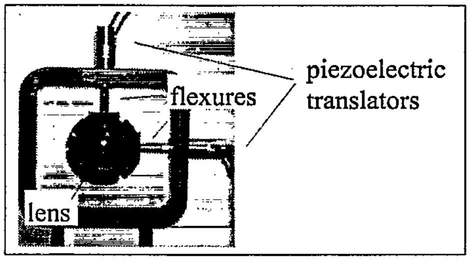


FIGURE 14 - Photograph of the microscanning head

The device was optimized to allow operation in real time up to a frame rate of 240Hz. Considering an integration time of 3 msec to build up the image, then one can estimate that each displacement including the time required for stabilization must last less than 1.17 msec. To achieve this level of performance, a special lens was designed to minimize the mass of the moving part and allow mounting close to the center of gravity. The lens was designed under contract by the National Optics Institute and the main characteristics are given in Table I.

TABLE I

MICROSCANNING LENS CHARACTERISTICS

FOV	6 deg.
Waveband	3-to 5-um
Aperture	f/3
Focal length	95 mm
Thickness	27 mm
Weight	110 g.
Material	silicon, germanium
# of elements	3
Blur spot	38 um at 5um

As shown on Fig. 14, the lens is mounted in the middle of a rigid frame and is held in place by two piezoelectric translators mounted perpendicularly. These translators provide the means to displace the lens on a plan perpendicular to the optical axis. The translators chosen for the application are from Physics Instrumente model P-910.625. Detailed specification is given in Table II.

TABLE II

SPECIFICATION OF THE PIEZOELECTRIC TRANSLATORS

	P-910.625
Nominal expansion at -1000V (μm)	40
Max. Pushing Force (N)	1000
Max. Pulling Force (N)	300
Stiffness (N/ μm)	30
Resonant Frequency (KHz)	11
Max. Operating Voltage (V)	-1000
Electrical capacitance (nF)	160
Temp. expansion (μm/K)	0.55

To get the flexibility required between the lens and the tip of the translator, flexible rods are used. Those were obtained from Physics Instrumente (model P-176.50) and are generally used to reduce the shear force in systems where both ends of a translator must be rigidly screwed. The rods measure 20 mm in length and are characterized by an axial stiffness at least three times larger than the stiffness of the translators (100 N/μm). Considering the geometry of the microscanning setup one can estimate the maximum shear force applied to be less than 0.2 N when the flexures are bent to maximum angle (540 μrad). This is small enough to prevent the breaking of the translators ceramics as a force of up to 1N can safely be applied.

To operate the microscanning head, each translator must be driven by a 0 to -1000 V power amplifier. The next section gives the description of an amplifier designed to fulfill that requirement. The description is given for one translator only, the same design is used to power up both channels.

3.2.2 The High Voltage Amplifier

To initiate the design of the high voltage amplifier, a rough estimate of the current needed to drive the translators and get the performances required for microscanning was obtained. The translators were modeled as pure capacitors (160nF) and the time required to perform a step was obtained from the standard charging equation of capacitors (Eq. 2).

$$V = \frac{1}{C}\int i\delta t \tag{2}$$

In Eq. 2, V stands for the the voltage applied to the translator, C its capacitance, i the charging current and t the time. Considering that the amplifier operates as a constant current source during a step, then Eq. 2 can be written as:

$$T = \frac{VC}{i} \tag{3}$$

Finally, assuming there is a linear relation between the translator length and the voltage applied then Eq. 3 can be modified as:

$$T = \frac{CLK}{i} \tag{4}$$

where L repsesents the step length and K, the expansion coefficient (25 V/um). Using Eq. 4, one can find that a current of at least 65 mA (i) is required to perform a 19um step (L) in less than 1.17 msec (T).

This condition is similar to the requirement for a microscanning step executed in the 2x2 mode at maximum frame rate. Of course, the calculation does not take into account the time required to stabilize the system. As a security factor, we designed the amplifier to operate at a higher current level.

The amplifier circuit is based on an chip from APEX (PA89). This device is specially designed to work with high voltage piezoelectric transducers. It is powered up by a source of ±600V and can deliver a current of up to 100mA in continuous mode. Higher peak currents can be obtained in dynamic applications as long as care is taken to stay within the safe operating conditions specified by the manufacturer.

In order to produce the large voltage swing required in our case (1000V), a master/slave bridge configuration was chosen. This configuration also has the advantage to double the slew rate of the amplifier. A complete drawing of the final circuit is shown in Appendix A. The device shown on the left hand part of the drawing is the master. It is configured with a gain of -50 and drives the slave device which is configured as an unity gain inverter. In a bridge configuration, the overall gain equals two times the gain of the master unit. It is thus possible to cover the entire output range with a single 0 to 10 V input signal.

To avoid oscillation of the output when connected to a large capacitive load as the translator, compensation was used to limit the bandwidth of the amplifier to 6.5KHz. To further improve the reliability of the amplifier, great care was taken in the development of protection circuitry. Firstly, low capacitance fast signal clamping diodes (1N4148) were mounted at the input of each PA89 device to prevent the differential input voltage from exceeding the maximum rated (±25V). The four diodes used in the design limit the differential input voltage to ±1.4 and allow sufficient voltage to drive the amplifier to maximum slew rate. Secondly, current limiting resistors were added both on the master and the slave amplifier. Actually, the output current is limited to 75mA however smaller limiting resistors could be used to increase the current available. The master current limitation has been set about 20% lower than the slave current limitation to allow equal sharing of the stress in the case of a fault such as a short across the load. Thirdly, fast feedback diodes (UF 4007) were added between the output and each alimentation pin to make sure that the output voltage never exceeds the alimentation. This could occur by inadvertently shocking the translator and induce a transient peak voltage that exceeds the admissible limit. Finally, voltage transient suppressors were added between each alimentation pin and the ground in order to make sure that each alimentation pin looks like a low impedance path for voltage transients.

All the circuitry shown on Appendix A was mounted on a small printed circuit board. Care was taken to minimize lead length and place the protection circuitry as close as possible to the PA89 devices.

The power required to operate the amplifier is obtained from a regulated high voltage DC module bought from Glassman High Voltage Inc. This unit is adjustable from 0 to ±1000V and can be turned on and off remotely using a TTL compatible signal.

3.2.2.1 Preliminary Results

The microscanning head was tested in conjunction with the high voltage amplifier to make sure it meets the specifications required for microscanning. Fig. 15 shows a curve of the output voltage versus time for a 250V step corresponding approximately to a 10um step.

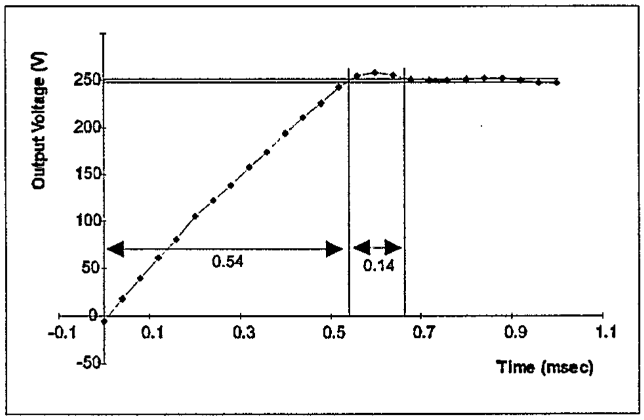


FIGURE 15 - The high voltage amplifier response time under load

As seen on Fig. 15, the amplifier response can be divided in two parts: a first where the amplifier responds with constant current (the voltage across the load increases linearly) and a second corresponding to the stabilization time. The first part last about 0.54 msec and is proportional to the step height. The second last approximately 0.14 msec and seems to be roughly independent of the step height. By extrapolation, it is possible to calculate the time required for a 19um step to be 1 msec plus the stabilization time (0.14 msec) which gives 1.14 msec. This is within the required specifications. Shorter displacement time will be possible by increasing the output current.

3.2.3 The Microscanning Controller

The microscanning controller serves two main purposes. Firstly, it synchronizes the displacements of the imaging lens to the FPA frame rate and integration time. Secondly, it sets the length and the direction of a microscanning step according to the mode selected.

Fig. 16 shows a schematic diagram of the controller and a detailed drawing is available in Appendix B. It is based on two microcontrollers of the INTEL family (87C752) working in synchronism. One is used for horizontal displacements while the other controls vertical displacements. Four digital input lines set the operation of the controller: the vertical sync input line, the microscan mode two bit input code and the reset line. In counterpart, the controller outputs two analog (0-10 V) and two digital signals. The analog outputs correspond to the X and Y position of the microscanning lens and are compatible with the high voltage amplifier described before. The two digital signals (MSSYNC and ENCNT) are used for synchronization purposes.

The operation of the controller can be initialized by pulling the reset line to a logic low level. Following this event, the memory of each microcontroller is loaded with a table corresponding to a microscanning pattern selected. Each location of the tables stores a coordinate expressed as a 12-bit integer. Then, each time a valid vertical synchronization occurs, a new coordinate is sent to a digital to analog converter which outputs a 0 to 10 volt signal compatible with the high voltage power amplifier.

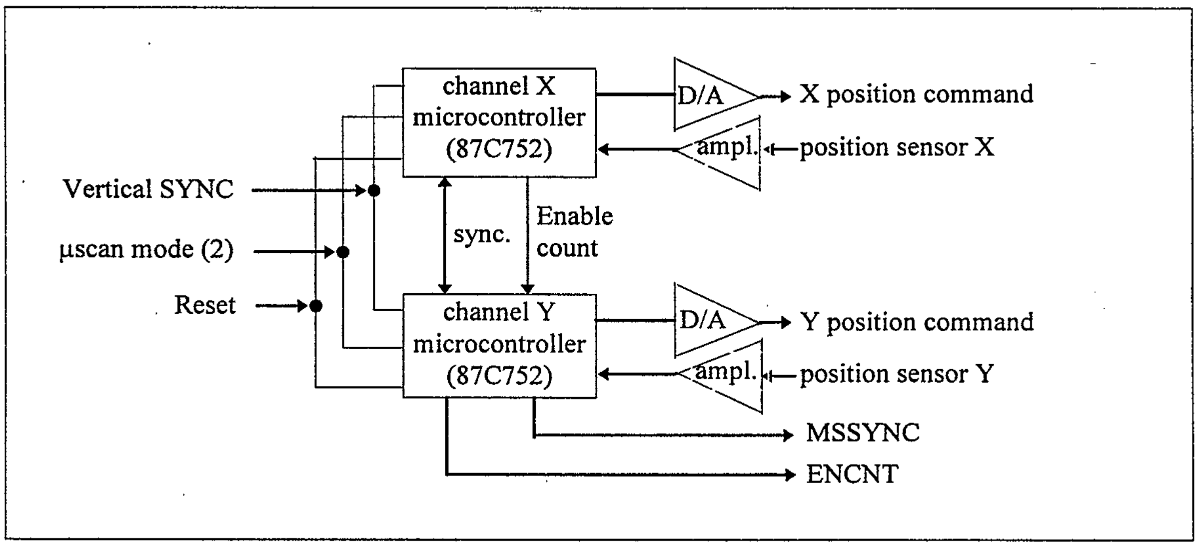


FIGURE 16 - Schematic diagram of the microscanning controller electronics

As mentioned before, the controller determines the step length and direction according to the mode selected. Four modes were implemented in the controller: no microscan, a 2x2 cycle, a 3x3 cycle and a 4x4 cycle. Fig. 17 shows the pattern associated with the last three cases. Position zero in each cycle corresponds to the first step position. When in this position, the MSSYNC signal is set to logic high level allowing synchronization with the microscanning processor (this unit will be described in section 3.2.5). Any microscanning pattern can be programmed in the controller memory. However, the 2x2 and 4x4 cycles chosen here are of interest as they are characterized by steps of equal size and require only one translator to work at a time.

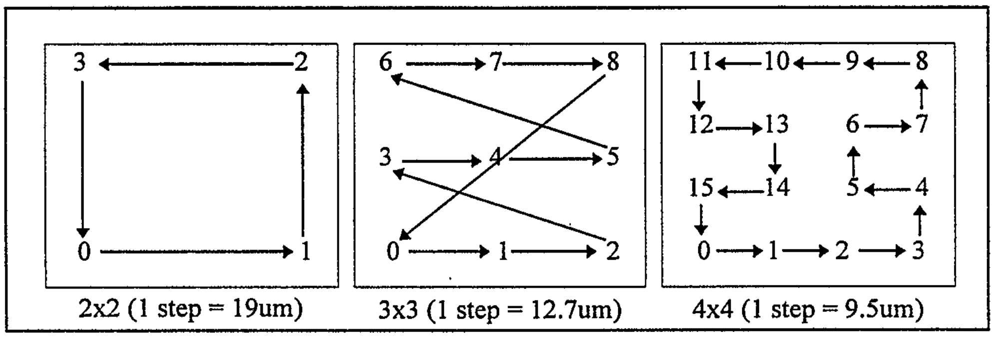


FIGURE 17 - The microscanning pattern according to the mode

Each time a step is done, an automatic calibration is performed to readjust the step length and minimize the positioning error. To allow this calibration, the signal from position sensors mounted on each translator is sent as a feedback to the microcontrollers. Before being digitized and used in the correction algorithm, the output of the position sensors is conditioned. The following sections give some details about the circuitry used.

3.2.3.1 Position Sensors

The sensors used to measure the position of the microscanning lens are based on strain gage devices connected in a bridge configuration. To accurately measure the output of the bridge, a differential amplifier is used. A schematic drawing of the amplifier designed is given in Appendix C. The bandwidth was limited to 3.5 KHz to minimize noise while still maintaining sufficient bandwidth to measure the fastest displacements. Potentiometers were added to allow gain calibration and adjustment of the zero point.

In order to set amplifier gain to the required value (0.25V/µm), a calibration system based on an interferometer was used. Fig. 18 shows a schematic drawing of the setup.

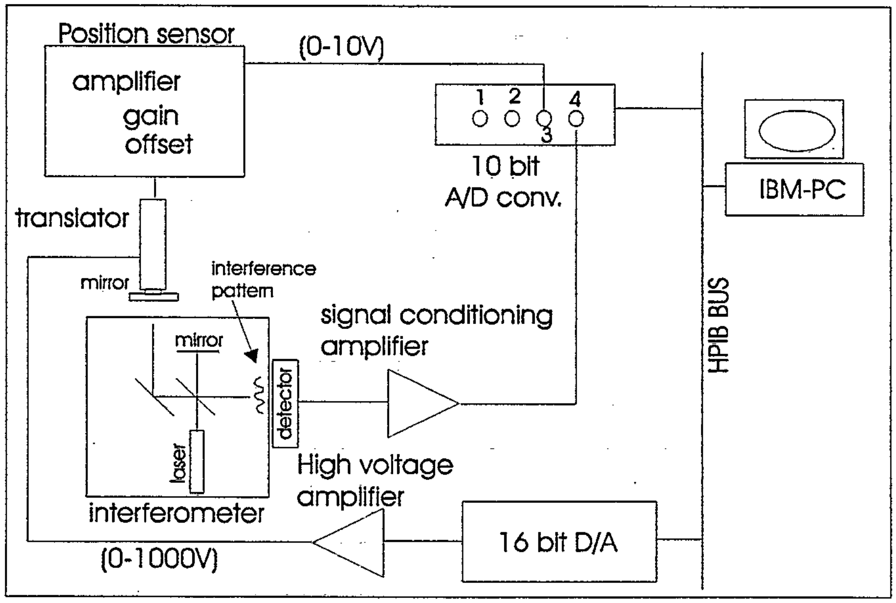


FIGURE 18 - Schematic drawing of the calibration setup

The system is mainly composed of a two channels 10 bits A/D converter and a 16 bit D/A converter which can be interfaced to an IBM-PC through an HPIB bus. The D/A converter is used to generate a precise voltage adjustable from 0 to 10 V. This signal feeds the input of a high voltage amplifier and drives a translator on which a flat mirror is mounted. The mirror and the translator have been aligned with the interferometer as to produce an interference pattern just in front of the detector plane. By varying the length of the translator, the phase of the interference pattern changes in direct proportion with the elongation. The output of the detector thus varies as a sinusoidal function having a period equals to half the wavelength of the laser used in the interferometer ($\lambda/2 = 316.4$ nm).

To calibrate the position sensors, the D/A converter was programmed as to generate a slow growing ramp and the output of the detector as well as the output of the position sensors were recorded with the A/D converter. Then, the number of periods recorded on the detector output was counted and the voltage variation was measured from the position sensor output. The gain of the amplifier was then calculated by multiplying the count measured by the period of the sinusoidal signal divided by the voltage required to obtain that count as shown in Eq. 5.

$$G = \frac{n(316.4nm)}{\Delta V} \tag{5}$$

In Eq. 5, n stands for the number of periods in the detector output and ΔV for the position sensor output voltage variation. The gain potentiometer was adjusted and the procedure repeated until the gain of each amplifier was calculated to be within less than 1% of the target value (0.25V/um).

Following the calibration procedure, some means must be introduced to allow the controller to correct for the rolling integration mode of the FPA. The next section gives a precise timing diagram of the controller and explains the method used to correct for that artifact.

3.2.3.2 Correction for Rolling Integration

The microscanning controller allows correction for the rolling integration mode of the FPA through the use of the ENCNT signal. The main use of this signal is to inform the microscanning processor that a valid image is available on the HSVB. The timing of the ENCNT signal takes into account the delay of one frame period imposed by the architecture of the Pro-View system (see section 3.1.2).

As mentioned before, the operation of the controller is based on the vertical synchronization signal provided by the Pro-View system ($\overline{FPA\_YSYNC}$). This signal corresponds to the frame timing of the first line of the FPA. When low, the detectors of that line integrate and the integration period ends upon a low to high transition of the signal which starts up the read out process. After completion, the signal remains high until a new integration period begins. As explained in section 2.5, the integration period of each line of the array is delayed by the time required to readout the preceding line. This creates an overlap in the integration period of successive images which is maximum when the integration time is set to the maximum value (see Fig. 12).

A timing diagram showing the relation between the FPA integration time and the ENCNT signal is given on Fig. 18.

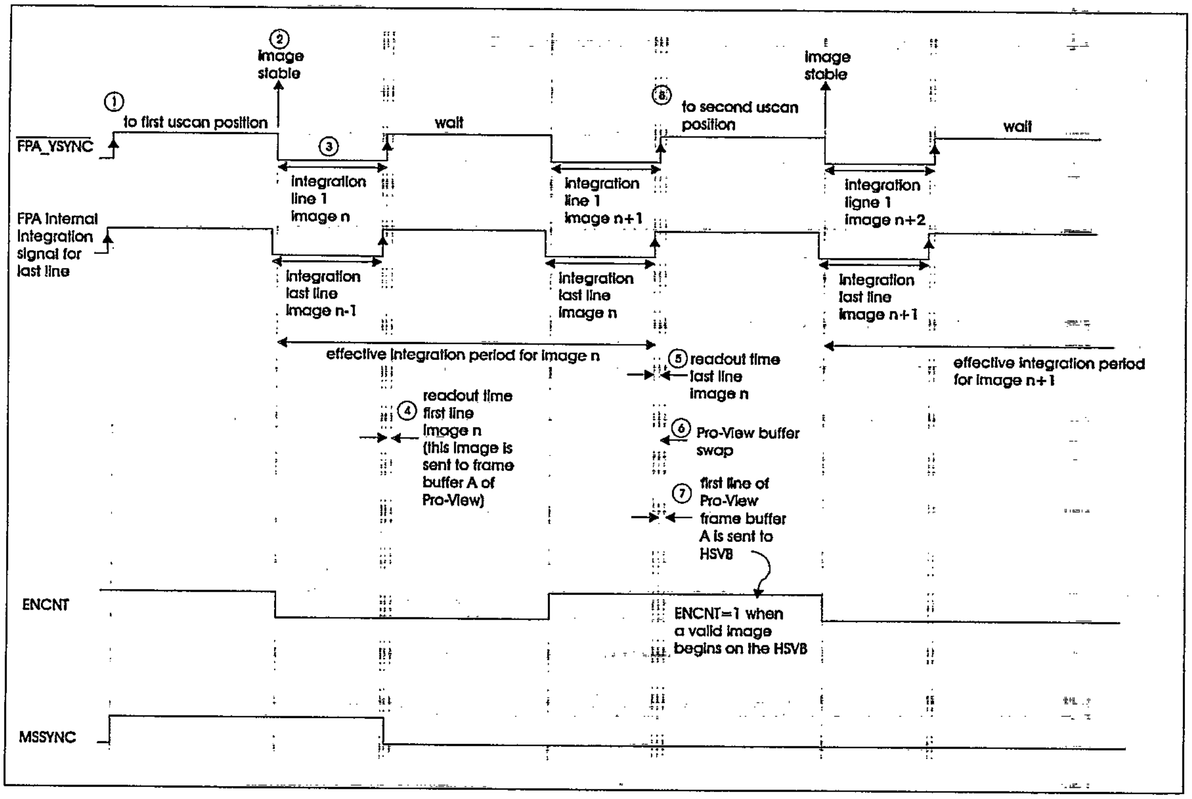


FIGURE 18 - Correction for rolling integration

To perform a valid microscan image acquisition sequence, the images must be stabilized over the FPA for the overall effective integration period of a frame. Thus, one image over two must be thrown away as it includes blurring effect due to displacement of the lens during signal integration. To implement this correction, the microcontrollers were programmed to count each second image and a signal (ENCNT) was generated to indicate every valid image.

The circled numbers on Fig. 18 give the sequence of events that occurs in the system. Firstly, the microscanning lens is brought to a first position (1). Then, the system waits for stabilization (2) and the integration time of the first line begins (3). Following this, the readout process of line one begins (4) and the image samples are sent to Pro-View frame buffer A. This process goes on until the last line of the FPA is read out (5). Then, a frame buffer swap occurs in Pro-View (6) and the data stored in frame buffer A is sent to the HSVB (7). At the same time, the lens is moved to the next microscanning position (8) and the process starts over again. As shown on Fig. 18, the ENCNT signal is high when a valid microscanning image begins on the HSVB.

The bottom part of Fig. 18 also shows the timing diagram of the MSSYNC signal. This signal is generated by the controller and indicates the beginning of a microscanning cycle. Both ENCNT and MSSYNC are used to control the operation of the microscanning processor and their use will be described later on.

3.2.3.3 The Software

The operation of the microscanning setup is under the supervision of the controller software. The same program is executed on both channels except that one includes the information for horizontal displacements while the other includes information for vertical displacements. Fig. 19 gives a schematic view of the algorithm used on one channel. As shown, it is made of a main routine and an interrupt routine. The detailed listing of the program is available in Appendix B.

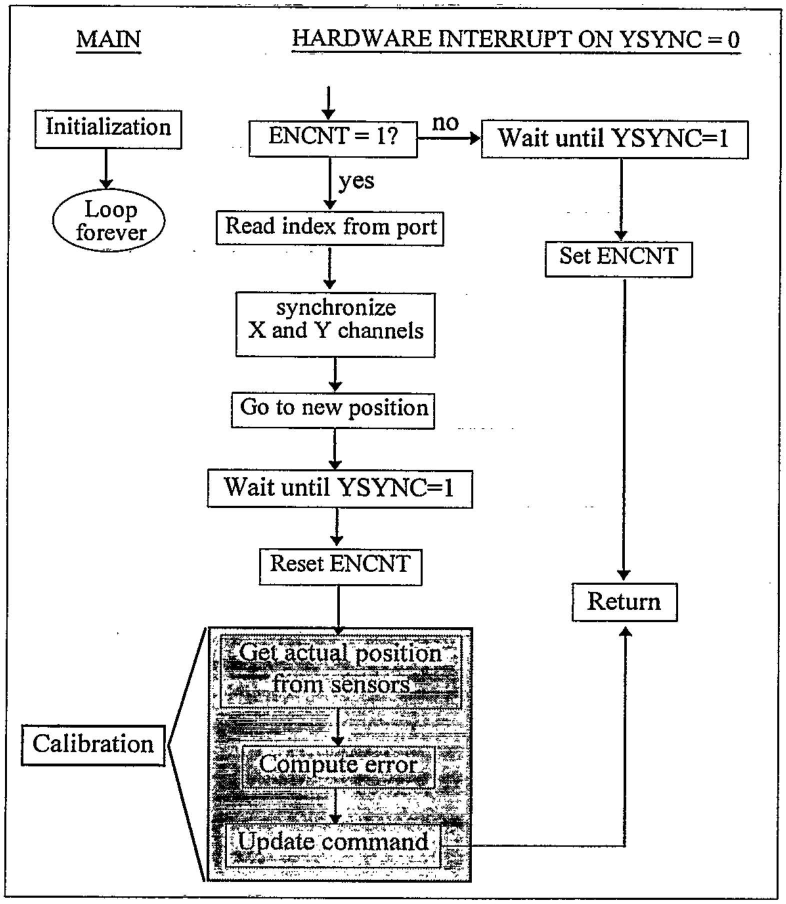


FIGURE 19 - Controller algorithm

The main routine is used to initialize the setup. It first determines the microscanning mode by reading the two-bit code (M1, M0) applied by the user to the mode input. This allows the access to a data table stored in ROM that defines the coordinates and the sequence of the microscanning cycle chosen. The table is then copied into RAM to be used as a voltage command table. This table defines the voltage levels required to bring the lens at each microscanning position. As it is stored in RAM, it is possible to update it to compensate for any positioning error. This procedure is done by the calibration algorithm and will be explained later on.

The last step of the initialization phase is to read the first location of the voltage command table and to bring the microscanning setup to its first position. Then, the program enters an infinite loop and waits for interrupts to occur.

The interrupt routine is entered on a low level of the inverted vertical synchronization signal (FPA_YSYNC). The first step executed is to determine the logic level of the ENCNT signal. If low, then the program waits for a transition of the FPA_YSYNC signal, sets ENCNT and returns to the main loop. Otherwise, the program determines the index of the next position in the microscanning cycle and then enters a synchronization phase. This is used to make sure that both channels operate in parallel. When ready, a new position sample is sent out and the processing is delayed until the FPA_YSYNC signal goes to logic high level. This phase corresponds to the time required to stabilize the lens.

After stabilization, the ENCNT signal is pulled low and the program enters the calibration phase represented as a gray box on Fig. 19. Calibration is required due to the non linearity of the translator response. In fact, the relation between the length of a translator and the voltage applied shows hysteresis which may introduce as much as 15% in positioning error. Of less importance are the temperature expansion change and the drift. The former can reach 0.2% per degree Kelvin while the latter is around 1% over one time decade. The calibration algorithm corrects the first two within the precision of the position measurements. The third is negligible considering the duration of each microscanning step.

The first step of the calibration process is to measure the position of the lens. An average of 16 8-bit samples are obtained from the position sensor. Then a positioning error (ε) is computed using Eq. 6.

$$\varepsilon = P - Rp(i) \tag{6}$$

where P represents the actual position and $Rp(i)$, the i^{th} element in the microscanning coordinate reference table stored in ROM. The error is then converted in voltage and used to update the voltage command table as shown in Eq. 7.

$$Vc_{t+1}(i) = Vc_t(i) + \varepsilon \tag{7}$$

where $Vc_{t+1}(i)$ represents the updated value of the i^{th} element of the voltage command table at time $t+1$. In our system, a one-to-one relation exist between the positionning error and the corresponding voltage due to the calibration. In fact, the position sensors are calibrated to produce a 0 to 10 volts output and the same voltage range is required to operate the high voltage amplifier. Following this calculation process, the program returns and waits for another interrupt to occur.

One iteration of the calibration algorithm is done each time the microscanning lens is displaced. Thus, to perform a full calibration, each position of a microscanning cycle needs to be visited a few times. Experience has shown that the positioning error comes down to negligible value after only a few iterations for each position. Then, the algorithm continues to track any variations due to the change in temperature.

The Fig. 20 shows an example of the calibration algorithm performance in the case of a 4x4 microscanning cycle. Fig. 20a) and b) give respectively a graph of the uncorrected and corrected cycle. A clear reduction in positioning errors is seen. The axis on Fig. 20 represents the amplitude of the voltage measured at the output of the horizontal and vertical position sensors calibrated in microns.

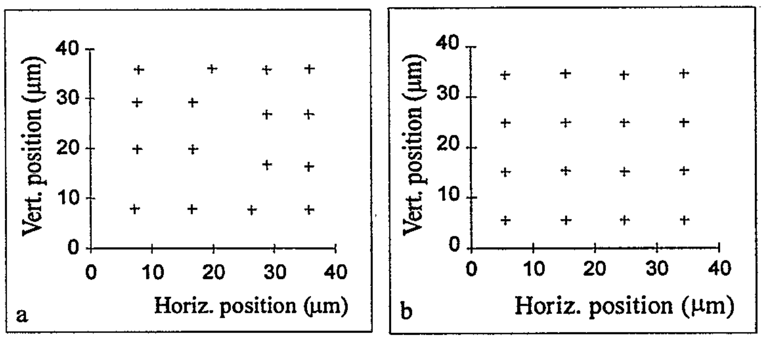


FIGURE 20 - 4x4 microscanning pattern: a) uncorrected b) corrected

3.2.4 The Microscanning Processor

The microscanning processor is used to build oversampled images in real time. It grabs a series of low-resolution images coming out on the HSVB, interlaces the pixels according to the microscanning mode selected by the user and outputs oversampled images compatible with the HSVB standard.

The operation of the processor is controlled by three logic signals: a two-bit code representing the microscanning mode (M1, M0) and the MSSYNC signal representing the beginning of a cycle. Those are obtained from the microscanning controller. The Fig. 21 gives a view of the processor operation in the case of the 4x4 cycle.

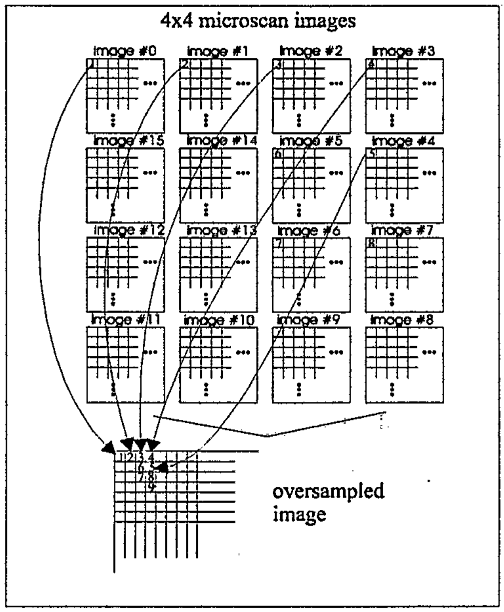


FIGURE 21 - The microscanning processor operation

The processor is structured as a dual frame buffer memory having different read and write sequences. The size of each frame buffer depends on the microscanning mode selected (M1, M0) and can reach a maximum of 1024x1024 pixels x 16 bit in the 4x4 mode. When one pixel is written in a frame buffer, another is read out from the other one. At the end of a cycle, the role of each buffer is exchanged and the process goes on. The beginning of a cycle is indicated by the level of the MSSYNC obtained from the controller.

During the write process, the address of a pixel is determined as a function of the image number within the microscanning pattern (see Fig. 21) in order to interlace the pixels properly. Any pattern could be used as long as the same is used by the processor and the controller. During the read cycle, the frame buffer memory is scanned linearly from top to bottom and the oversampled image is sent to the output. However, to compensate for the images dropped during the acquisition process (rolling integration), the output frame buffer can be scanned twice while one image is built in the input frame buffer.

The architecture of the processor is based on the Complex Programmable Logic Device (CPLD) technology of Intel. Two iFx780-15 chips are used to control the operation of the two frame buffers. A detailed schematic drawing of the processor is given in Appendix D. The two CPLD are programmed as row and column counters synchronized with the HSVB horizontal and vertical synchronization signals. They provide addresses for the frame buffers and the counting sequence depends if the buffer is in the writing or the reading mode. To implement this algorithm in the CPLD, two independent sets of counters were programmed in each CPLD. The output of each counter was tied together and to prevent bus contention, only one counter output is enable at a time. The selection depends if the frame buffer in the read or the write mode.

Before using the processor, the two CPLD must be programmed. The same program is used for mode 0, 1 and 3 (no microscan, 2x2 and 4x4) while a different file is required for mode 2 (3x3). Table III gives the name of the source file that must be used in each case and Appendix D gives a detailed listing of each file.

TABLE III) Source file required to program the CPLD as a function of the microscanning mode.

Microscanning mode	Source file name
no microscan, 2x2, 4x4	U1: vidu1_h.pds
	U11: vidu11_h.pds
3x3	U1: v33u1_l.pds
	U11: v33u11_l.pds

The source files must be compiled into JEDEC files before being downloaded to the CPLD. This is accomplish by using a compiler (PLDshell.exe) available from the CPLD vendor. Then a downloader software (pengn.exe) and a hardware interface cable must be used to terminate the procedure. Complete explanations are given in "PLDshell Plus/PLDasm User's Guide V3.1" (Ref. 16).

3.3 The Video Unit

The video unit is the last stage of the microscanning setup. It is used to grab the images produced by the microscanning processor and to display them live. It is composed of a digital frame grabber manufactured by Dipix Inc. and a standard high resolution video monitor. The frame grabber interfaces directly to the microscanning processor digital output and can be programmed to grab and display even at maximum frame rate (32Mb/sec). To provide for the four microscanning modes, distinct camera configuration files must be used. Appendix E gives a listing of the file required in each case.

4.0 RESULTS

To evaluate the performance of the microscanning setup, the LSF and the MDTF of the microscanning camera were measured and compared with the LSF and MDTF of the non-microscanning camera. Moreover, the mechanical characteristics of the setup were evaluated by tracing a graph of the MDTF as a function of the camera frame rate. The next three sections give a summary of the results obtained.

4.1. The Non-microscanning Case

The LSF and MDTF of the non-microscanning camera were obtained. To realize this measurement, the operating parameters of the camera controller were set to typical values. The integration time was set to 3 ms with a pixel clock of 5 MHz. This clock rate produces a frame rate of 72.88 Hz. The gain and offset of the camera (transimpedance amplifier) were adjusted to allow a two-point calibration using blackbodies of 20 °C and 30 °C. After calibration, the camera was aligned in the optical path of a collimator having a focal length of 120 inches to look at a narrow slit. The slit width was chosen to subtend an angle more than ten times smaller than the angle subtended by a single detector (25 μrad vs 400 μrad). It was slightly tilted to cover at least two columns of detectors and the temperature of the blackbody placed behind the slit was set around 120 °C in order to get enough signal without saturating the detectors.

Fig. 22 shows a region of the image recorded. A zooming factor of four was applied to facilitate comparison with the microscanning image.

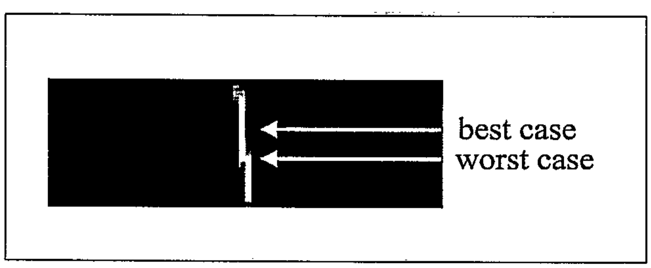


FIGURE 22 - The non-microscanning image of the slit

Fig. 23a gives the LSF of the non-microscanning FPA. For clarity, only two profile lines were overlaid on the graph. They represent the best and the worst cases as shown by the arrows on Fig. 22. The MDTF was obtained by calculating the Fourier transform of the LSFs. Results are shown in Fig. 23b. Many curves were plotted corresponding to positions between the best and the worst case. Considering that the focal length of the lens is 95 mm and that the detector pitch is 38 μm, then it is normal to find the first zero of the MDTF at a frequency of 1.25 cy/mrad. The use of microscanning should bring that limit close to 2.5 cy/mrad.

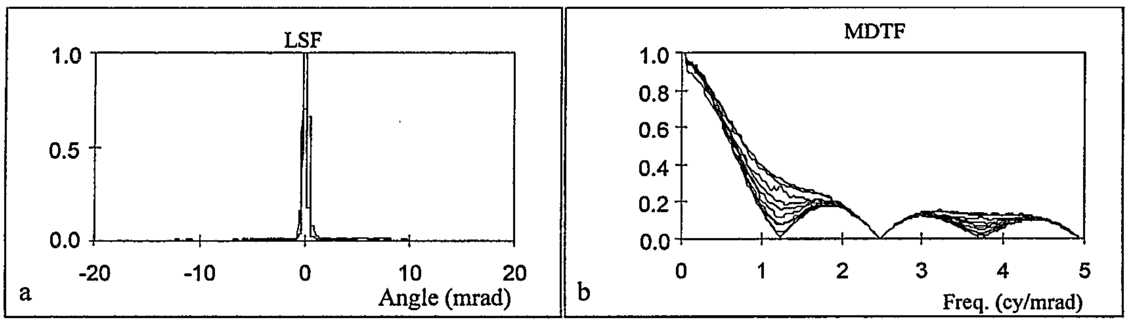


FIGURE 23 - Non-microscanning FPA a) LSF, b) MDTF

4.2 The 4x4 Microscanning Case

The LSF and MDTF of the microscanning camera were obtained using the same camera setting with the microscanning switch set to the 4x4 mode. Fig. 24 gives an example of the image obtained. The corresponding LSF curve is plotted on Fig. 25a. Unlike the preceding case, no difference is observed between each line profile. For that reason Fig. 25a) shows only one LSF curve.

FIGURE 24 - The microscanning image of the slit

The MDTF data was obtained using the same method as before. Results are given on the graph of Fig. 25b. All the MDTF curves show superposed. It is possible to see that the first zero of the function occurs around 2.5 cy/mrad which is the limit predicted by the theory.

As seen on Fig. 25b, the aliasing lobe of the MDTF is attenuated as compared to the theoritical case presented on Fig. 5. This is due to a smoother LSF than what is predicted by theory and can be attributed to diffraction. As generally known, the microscanning lens is at best limited by diffraction. Considering a wavelength of 4um and the aperture (f/#) of the microscanning lens then the diffraction theorem stipulates that almost all the energy coming from a point source should be contained in a disc of nearly 30um radius. This almost corresponds to the pixel size (31um) of the FPA and could easily explain the degradation measured in the LSF curve.

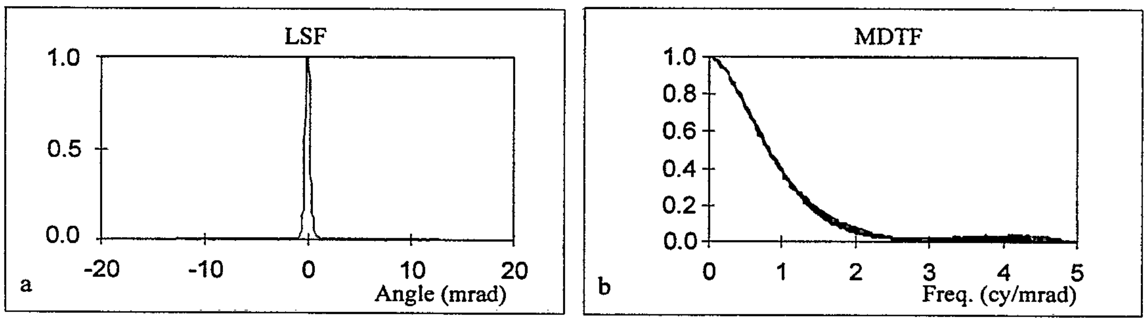


FIGURE 25 - Microscanning FPA a) LSF, b) MDTF

As another way to evaluate the performance of the microscanning system, the image of a standard test pattern was obtained using the same camera parameters as before. The test target was adapted to the detector size and the lens used and shows frequencies increasing from about 0.5 cy/mrad up to approximately 3 cy/mrad in the case of the horizontal pattern and 5 cy/mrad for the vertical and diagonal ones. The left hand part of Fig. 26 shows the image obtained without microscan and the right hand part shows the result obtained in the 4x4 microscanning mode. The image of Fig. 26a was zoomed four times in order to show the same dimension as the microscanned image.

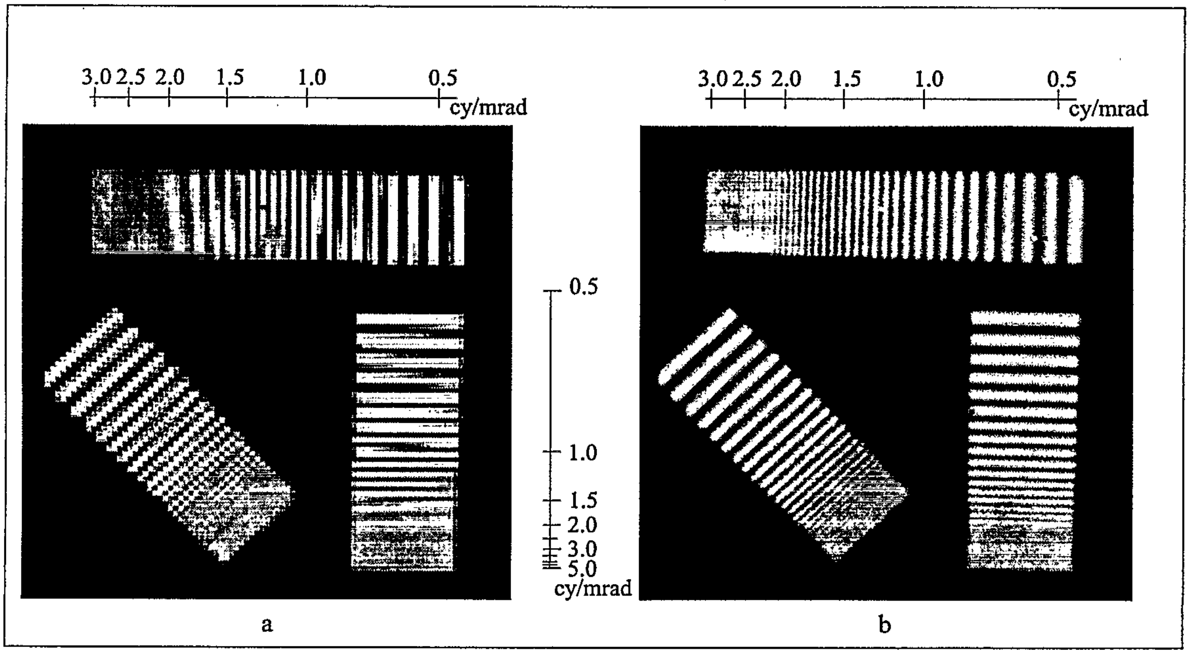


FIGURE 26 - Image of a standard test pattern: a) without microscanning b) with a 4x4 microscanning pattern

As shown on Fig 26a, the non-microscanned camera can resolve frequencies up to 1.25 cy/mrad. Frequencies seen above that limit are false frequencies and are due to aliazing and undersampling of the FPA. On Fig. 26b, enough contrast still remains to resolve frequencies up to 2.5 cy/mrad. No aliasing is observed.

4.3 Frequency Response

As a method to evaluate the mechanical performance of the setup, the MDTF of the microscanning camera was plotted as a function of the camera frame rate. For this experiment, the integration time was fixed to 3msec and the frame rate was varied from about 10 Hz to more than 200Hz with the microscanning system set in the 4x4 mode. A two-point calibration was performed for each frequency tested and a MDTF measurement was performed each time.

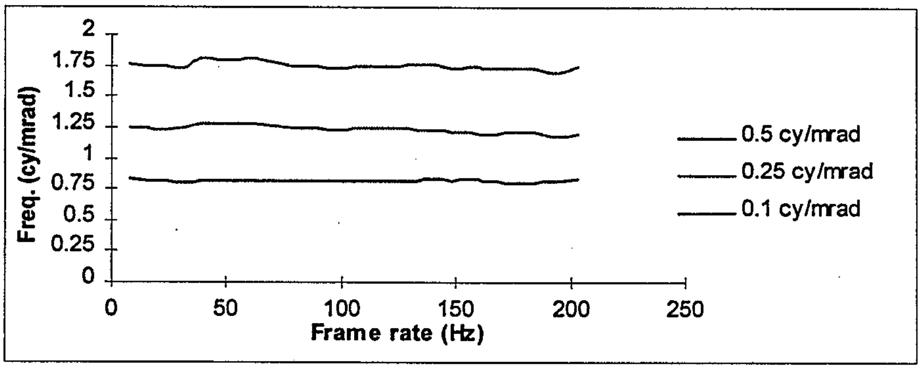


FIGURE 27 - MDTF as a function of the microscanning camera frame rate

Fig. 27 shows the MDTF height at 0.5, 0.25 and 0.1 cy/mrad as a function of the camera frame rate. We can clearly see that the MDTF is not affected by a change of the camera frame rate over the whole frequency range tested. We can thus conclude that the microscanning device meets the requirements of the project in terms of frame rate.

5.0 CONCLUSION

This memorandum describes work performed to develop and test a microscanning camera system. The system built allows to improve the resolution of a FPA detector matrix by a factor of two and can be operated with a frame rate of up to 240 Hz. The microscanning effect is obtained by displacements of a lens placed in front of a focal plane array. In order to achieve the high performance required, a new method based on a two-axis microtranslation device is used. This device allows very compact fabrication and operation at high frame rate.

The system includes all the hardware and software necessary to implement the microscanning operation with real-time image processing. The system can be operated in a 2x2, 3x3 and 4x4 microscanning mode and the pattern can be changed by software control.

The operation of the system was thoroughly tested and results show a clear improvement of the maximum resolvable frequency. Moreover, tests made over a wide camera frame rate range showed no degradation of the microscanning images overall the frequency range.

Future work could be done to further improve the resolution of microscanned images. On one hand, filtering could be used to enhance the contrast over the resolved band. As shown by the results, the MDTF of microscanned images decreases gradually up to the maximum resolvable frequency. To ease the detection and identification process of objects including high frequencies, one could try to develop a filter matched to the MDTF curve and enhance the contrast of the highest frequencies. On the other hand, superresolution algorithms could be developed to exploit every sample of a microscanning cycle and bring the resolution of superresolved images close to the size of a microscanning step.

6.0 REFERENCES

1. Pratt W.K., "Digital signal processing", John Wiley & Sons, New-York, N.Y., 1978.

2. Bradley D.J. and Dennis P.N.J., "Sampling effects in CdHgTe focal plane arrays", SPIE Vol. 590 Infrared Technology and Applications, 1985, pp. 53-60.

3. Irami M. And Peleg S., "Improving resolution by image registration", Computer Vision, Graphics Image Processing: Graphical Models and Image Processing, Vol. 53, No 3, 1991, pp. 231-239.

4. Wolfe W.L. and Zissis G.J., "The infrared handbook", Infrared Information Analysis Center, Washington DC, Revised edition 1985.

5. Schuberth W., "Characterization of the spatial resolution of focal plane arrays", Proposal for the RSG.16 Subgroup on Focal Plane Arrays, Forschungsinstitut für Optik, Tübingen, 1991.

6. Schuberth W., "Measurement procedures for characterization of the spatial resolution of focal plane arrays", Proposal for the RSG.16 Subgroup on Focal Plane Arrays, 1991.

7. Awamoto K., Ito Y., Ishizaki H. And Yoshida Y., "Resolution improvement for HgCdTe IRCCD", SPIE Vol. 1685 Infrared Detectors and Focal Plane Arrays II, 1992, pp. 213-220.

8. Blommel F.P., Dennis P.N.J. and Bradley D.J., "The effects of microscan on starring infrared sensor imagery", SPIE Vol. 1540, Infrared Technology XVII, 1991, pp. 653-664.

9. Barnard K.J., Watson E.A. and McManamon P.F., "Nonmechanical microscanning using optical space-fed phased arrays", SPIE Vol. 2224 Infrared Imaging Systems: Design, Analysis, Modeling and Testing V, 1994, pp. 168-179.

10. Fortin J. and Chevrette P., "Realization of a fast microscanning device for focal plane arrays", SPIE Vol. 2743 Infrared Imaging Systems: Design, Analysis, Modeling and Testing VII, 1996, pp. 185-196.

11. Jacquemod G., Odet C. and Goutte R., "Superresolution technique applied to CCD sensors", Signal Processing IV: Theories and Applications, 1988, pp. 847-850.

12. Jacquemod G., Odet C. and Goutte R., "Image resolution enhancement using subpixel camera displacement", Signal Processing, Vol. 26, 1992, pp. 139-146.

13. Fortin J. and Chevrette P., "Infrared Eye: Evaluation of an Infrared Focal Plane Array Camera System (AE4256)", DREV TM-9413, July 1994, Unclassified.

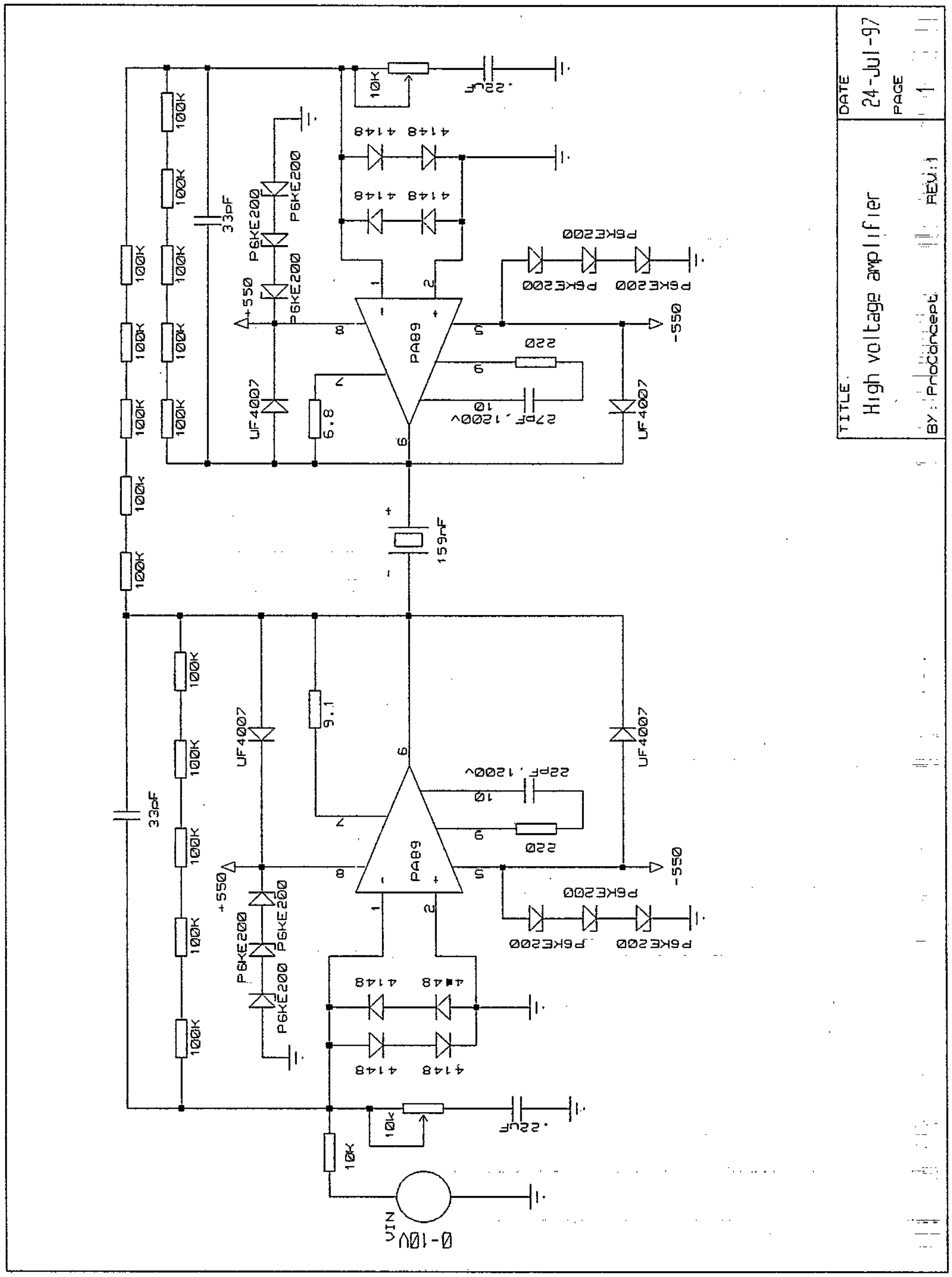
14. "Amber Engineering Pro-View hardware manual", Release 1.1, Amber Engineering Inc., 1991.

15. "Real-time Application system (AE-RTAS) Users' Guide", Release 1.1, Amber Engineering Inc., 1991.

16. "PLDshell Plus/PLDasm User's Guide V 3.1", Intel Corporation, Santa Clara, CA, 1991.

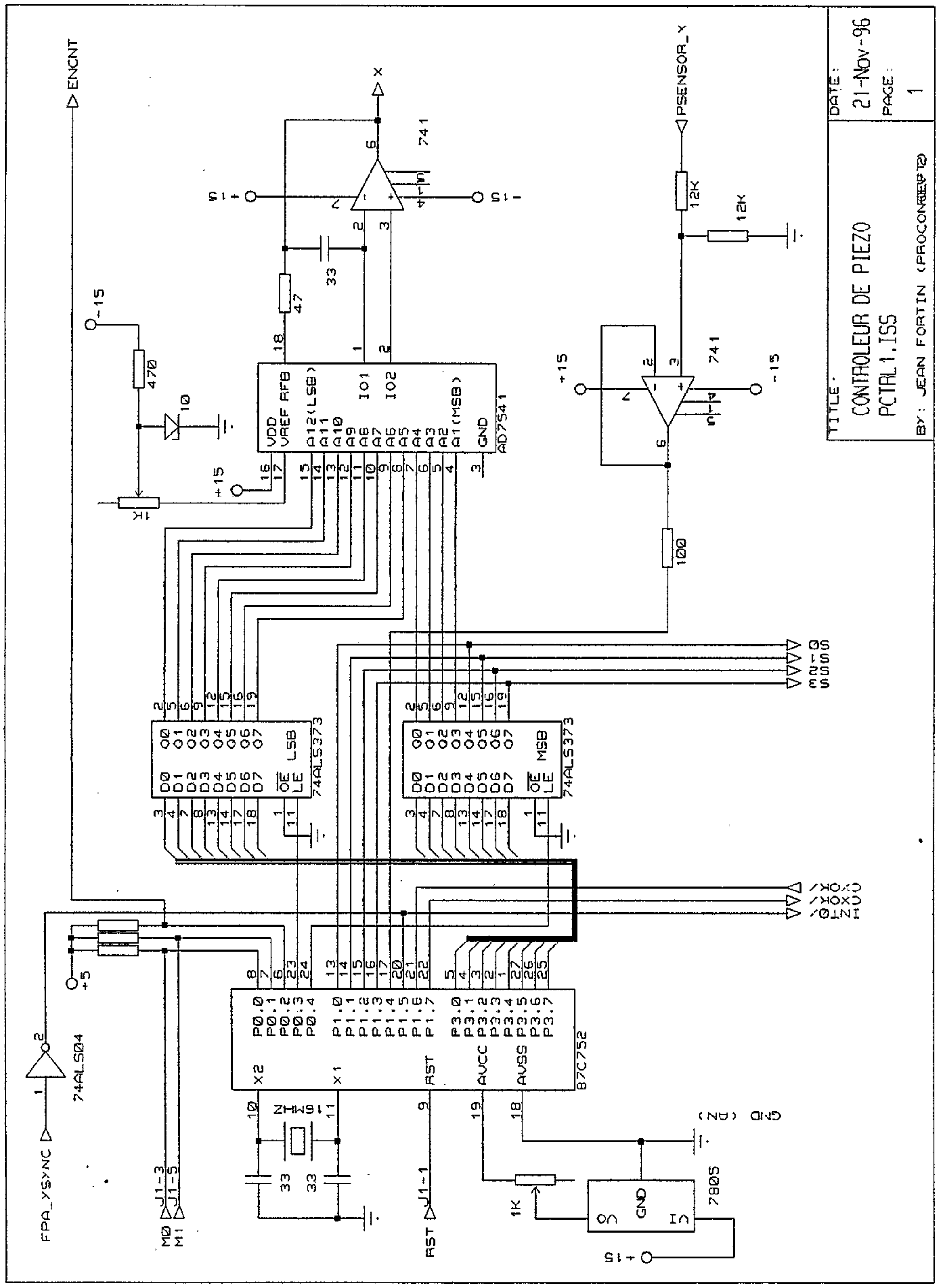
APPENDIX A

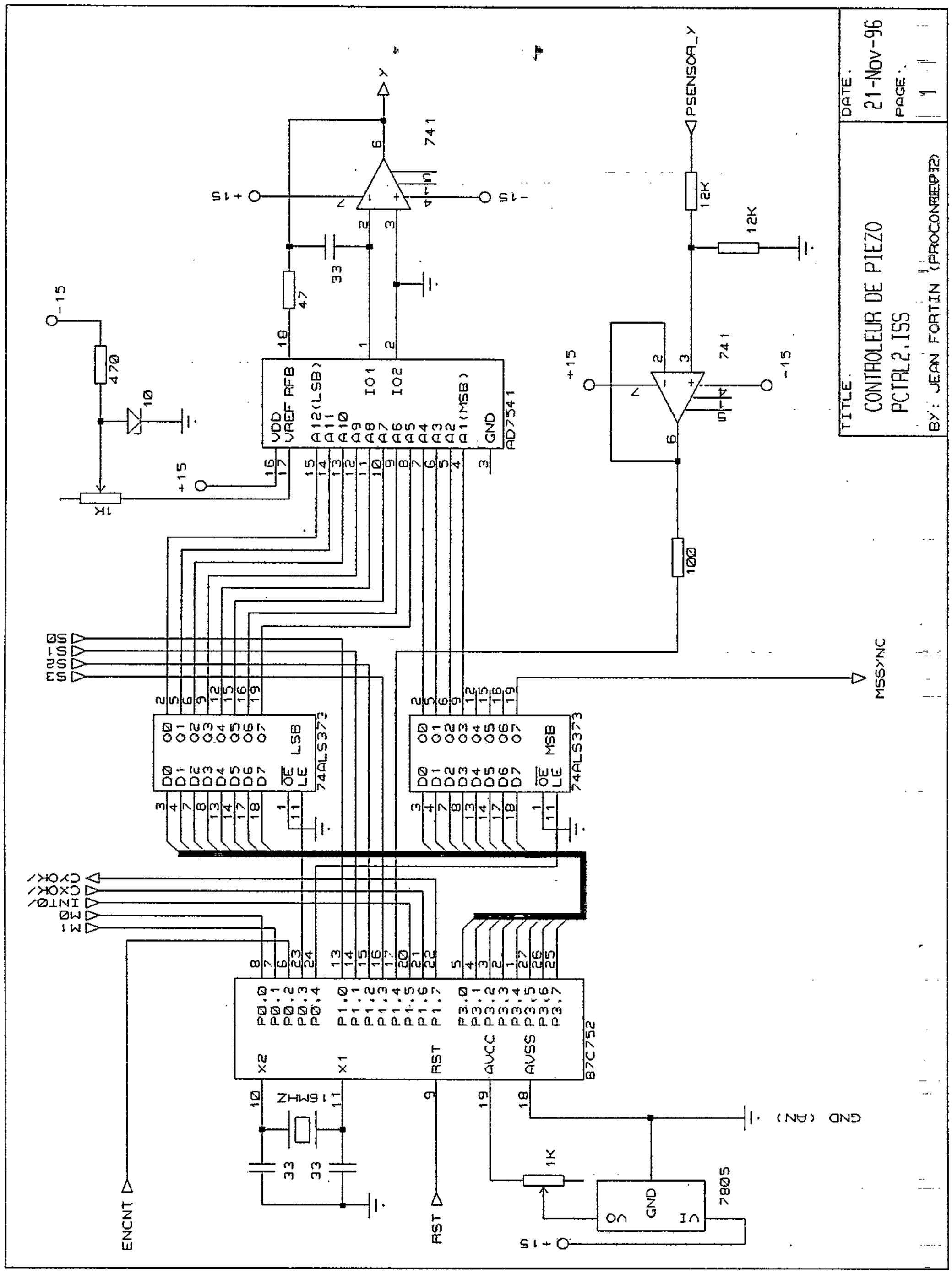
High Voltage Power Amplifier Schematics



APPENDIX B

Schematic Diagram of the Microscanning Controller




TITLE:
CONTROLEUR DE PIEZO
PCTRL2.ISS
BY: JEAN FORTIN
DATE:
21-Nov-96
PAGE:
1
AD7541
74ALS373
87C752
741
7805
PSENSOR_Y
MSSYNC
ENCNT
RST
GND (AN)

```
;
;  PCTRL_X.ASM

; Version 2: modifie pour sequential scan integration

;  Primary controls
$MOD752
$TITLE (piezo control program)
$PAGEWIDTH(132)
$DEBUG
$OBJECT
$NOPAGING


;****************************************
;       VECTOR TABLE DEFINITION
;****************************************
;

; RESET CONDITION
        ORG     0000H
        AJMP    MAIN

; EXTERNAL INTERRUPT #0
        ORG     0003H
        AJMP    INT_0

; TIMER FLAG 0
        ORG     000BH
        RETI

; EXTERNAL INTERRUPT #1
        ORG     0013H
        RETI

; IIC TIMER OVERFLOW
        ORG     001BH
        RETI

; SERIAL PORT INTERRUPT
        ORG     0023H
        RETI

; ADC EOC
        ORG     002BH
        RETI

; PWM COUNTER OVERFLOW
        ORG     0033H
        RETI


;****************************************
;        DATA TABLE
;****************************************
;
        ORG     0040H

M1X1:
        DW      00800H

M2X2:
        DW      02433H, 00BCDH
        DW      03433H, 01BCDH

M3X3:
        DW      012EFH, 02800H, 06D11H
        DW      042EFH, 05800H, 00D11H
        DW      072EFH, 08800H, 03D11H

M4X4:
        DW      0424DH, 0061AH, 019E6H, 02DB3H
        DW      0524DH, 0961AH, 079E6H, 03DB3H
        DW      0C24DH, 0861AH, 069E6H, 0ADB3H
        DW      0D24DH, 0E61AH, 0F9E6H, 0BDB3H

;4x4 lineaire
;          DW      0124DH, 0261AH, 039E6H, 0CDB3H
;          DW      0524DH, 0661AH, 079E6H, 00DB3H
;          DW      0924DH, 0A61AH, 0B9E6H, 04DB3H
;          DW      0D24DH, 0E61AH, 0F9E6H, 08DB3H


T_LENGTH:
        INC     A
        MOVC    A, @A+PC
        RET
        DB      02H, 08H, 012H, 020H


TABLE   EQU     020H

;***************************************************************
;
;       MAIN PROGRAM
;
;       R7 CONTIENT L'OFFSET.
;
;       DPTR POINTE AU DEBUT DE LA TABLE COURANTE (ROM).
;
```

```
;***************************************************************
;
        ORG     100H

MAIN:

        CLR     P0.3            ; CLEAR LATCH ENABLE ON LSB CHIP
        CLR     P0.4            ; CLEAR LATCH ENABLE ON MSB CHIP
        SETB    P1.7            ; CXOK/ = 1

        SETB    P0.2            ; ENCNT = 1

        MOV     ADCON, #0E4H    ; ENABLE ADC CHANNEL #4

;
; DETERMINE THE MICROSCANNING MODE
;
        MOV     A, P0           ; READ DIP SWITCH
        ANL     A, #00000011B   ; APPLY MODE MASK


        CJNE    A, #00H, M2     ; SET DPTR TO THE BEGINNING OF
        MOV     DPTR, #M1X1     ; CORRESPONDING TABLE
        SJMP    DPTR_OK
M2:     CJNE    A, #01H, M3
        MOV     DPTR, #M2X2
        SJMP    DPTR_OK
M3:     CJNE    A, #02H, M4
        MOV     DPTR, #M3X3
        SJMP    DPTR_OK
M4:     MOV     DPTR, #M4X4
DPTR_OK:

;
; TRANSFER COMMAND TABLE IN RAM
;
        ACALL   T_LENGTH        ; DETERMINE THE SIZE OF THE TABLE
        MOV     R2, A           ; SAVE TABLE LENGTH
        MOV     R7, #00H        ; SET OFFSET TO 0
        MOV     R0, #TABLE
L1:     MOV     A, R7
        MOVC    A, @A+DPTR
        MOV     @R0, A
        INC     R0
        INC     R7
        DJNZ    R2, L1


        MOV     R7, #00H        ; INITIALIZE OFFSET TO 00H


;
; WRITE FIRST SAMPLE TO OUTPUT PORT
;
        ACALL   READ_RAM_TABLE  ; READ FIRST SAMPLE FROM TABLE
        ACALL   WRITE_PORT      ; WRITE SAMPLE TO OUTPUT PORT

        MOV     IE, #10000001B  ; ENABLE INTERRUPTS

LOOP:
        AJMP    LOOP

;***************************************************************
;
; WRITE_PORT SUBROUTINE
;       INPUT:  R2 R3
;
;***************************************************************
WRITE_PORT:
        MOV     P3, R2          ; WRITE MSB
        SETB    P0.4
        CLR     P0.4
        MOV     P3, R3          ; WRITE LSB
        SETB    P0.3
        CLR     P0.3
        RET


;***************************************************************
;
; READ_RAM_TABLE SUBROUTINE
;       INPUT:  R7
;       OUTPUT: R2 R3
;       ACC AND R0 ARE MODIFIED
;
;***************************************************************
READ_RAM_TABLE:
        MOV     A, #TABLE
        ADD     A, R7
        MOV     R0, A
        MOV     A, @R0
        MOV     R2, A
        INC     R0
        MOV     A, @R0
        MOV     R3, A
        RET


;***************************************************************
;
; WRITE_RAM_TABLE SUBROUTINE
```

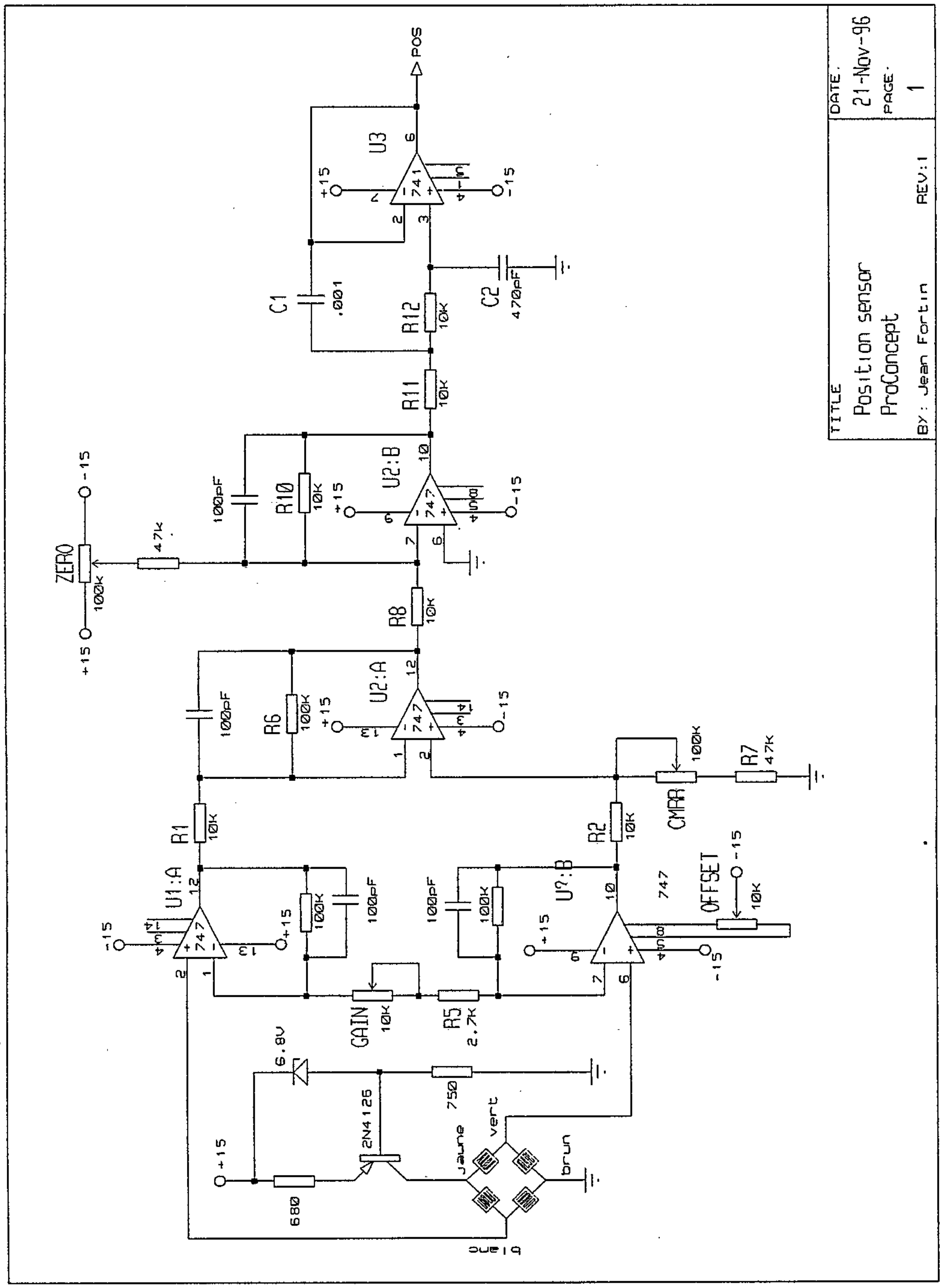
```
;       INPUT:  R7 R2 R3
;
;       ACC AND R0 ARE MODIFIED
;
;***************************************************************
WRITE_RAM_TABLE:
      MOV     A, #TABLE
      ADD     A, R7
      MOV     R0, A
      MOV     A, R2
      MOV     @R0, A
      INC     R0
      MOV     A, R3
      MOV     @R0, A
      RET

;***************************************************************
;
; READ_ROM_TABLE SUBROUTINE
;       INPUT:  R7, DPTR
;       OUTPUT: R4 R5
;
;       ACC IS MODIFIED
;
;***************************************************************
READ_ROM_TABLE:
      MOV     A, R7
      MOVC    A, @A+DPTR
      MOV     R4, A
      MOV     A, R7
      INC     A
      MOVC    A, @A+DPTR
      MOV     R5, A
      RET

;***************************************************************
;
; INT0 SUBROUTINE (LEVEL TRIGGERED)
;       ACC, R0, R2, R3 ARE MODIFIED
;
;***************************************************************

INT_0:

        JB      P0.2, UPDATE    ; Check ENCNT and update position
if 1

W2:     JNB     P1.5, W2        ; WAIT UNTIL PIN 20 GOES HIGH

      SETB    P0.2            ; ENCNT = 1
      RETI            ; TERMINATE INTERRUPT.

;
; GO TO NEXT MICROSCANNING STEP AS FAST AS POSSIBLE
;
UPDATE:
        MOV     A, P1           ; GET THE NEW OFFSET VALUE FROM
PORT 1
      ANL     A, #0FH
      RL      A               ; MULTIPLY BY 2
      MOV     R7, A           ; SAVE OFFSET VALUE

      CLR     P1.7            ; SYNCHRONIZE X AND Y CHANNELS
W1:     JB      P1.6, W1
      SETB    P1.7

      ACALL   READ_RAM_TABLE
      ACALL   WRITE_PORT

W3:     JNB     P1.5, W3        ; WAIT UNTIL PIN 20 GOES HIGH
      MOV   A, P0 ; CHECK USCAN MODE AND DO NOT CLEAR ENCNT
      ANL   A, #00000011      ;    IF USCAN MODE=0
      JZ    CAL
      CLR     P0.2            ; ENCNT = 0

;
; THE CALIBRATION ROUTINE IS EXECUTED DURING THE INTEGRATION TIME.
; A LOT OF TIME IS THUS AVAILABLE.
;

CAL:
      ACALL   SAMPLE          ; READ CURRENT POSITION (R2, R3)

      ACALL   READ_ROM_TABLE  ; GET IDEAL POSITION (R4, R5) AND
      MOV     A, R4           ; CLEAR INDEX BIT OF R4
      ANL     A, #0FH
      MOV     R4, A

      CLR     C               ; COMPUTE ERROR (R4R5 = R4R5-R2R3)
      MOV     A, R5
      SUBB    A, R3
      MOV     R5, A
      MOV     A, R4
      SUBB    A, R2
      MOV     R4, A

      ACALL   READ_RAM_TABLE  ; ADJUST COMMAND IN RAM TABLE
      MOV     A, R2           ; SAVE TABLE INDEX FOR FUTURE USE
      ANL     A, #0F0H
      PUSH    ACC
      MOV     A, R2           ; CLEAR INDEX BITS OF R2
      ANL     A, #0FH
      MOV     R2, A

      MOV     A, R3           ; ADD ERROR (R2R3 = R2R3 + R4R5)
      ADD     A, R5
      MOV     R3,A
      MOV     A, R2
      ADDC    A, R4
      MOV     R2, A

      MOV     A, R2           ; CHECK R2R3 FOR OUT OF RANGE VALUE
      ANL     A, #0F0H
      JZ      RANGE_OK
      ANL     A, #080H
      JZ      OVER
      MOV     R2, #00H
      MOV     R3, #00H
      AJMP    RANGE_OK
OVER:   MOV     R2, #0FH
      MOV     R3, #0FFH

RANGE_OK:
      POP     ACC             ; RESTORE INDEX BITS IN R2
      ORL     A, R2
      MOV     R2, A

      ACALL   WRITE_RAM_TABLE ; SAVE MODIFIED VALUE IN RAM TABLE

INT0_END:
      RETI

;***************************************************************
;
; SAMPLING ROUTINE
;       THE VALUE IS RETURNED IN R2(MSB) AND R3(LSB)
;       ACC AND R4 ARE MODIFIED.
;
;***************************************************************
SAMPLE:
      MOV     R2, #00H        ; INITIALIZE RESULT
      MOV     R3, #00H

      MOV     R4, #010H       ; SUM 16 SAMPLES

      MOV     A, ADAT         ; CLEAR ADCI

START_ADC:
      MOV     A, ADCON        ; START CONVERSION
      ORL     A, #08H
      MOV     ADCON, A

WAIT_EOC:                       ; WAIT UNTIL END OF CONVERSION
      MOV     A, ADCON
      ANL     A, #010H
      JZ      WAIT_EOC

      MOV     A, ADAT         ; READ RESULT

      ADD     A, R3           ; ADD RESULT
      MOV     R3, A
      CLR     A
      ADDC    A, R2
      MOV     R2, A

      DJNZ    R4, START_ADC   ; LOOP R4 TIMES

      RET

      END
```

```
;
;  PCTRL_Y.ASM

; Version 2: modifiee pour sequential scan integration

;  Primary controls
$MOD752
$TITLE (piezo control program)
$PAGEWIDTH(132)
$DEBUG
$OBJECT
$NOPAGING


;****************************************
;       VECTOR TABLE DEFINITION
;****************************************
;

; RESET CONDITION
      ORG      0000H
      AJMP     MAIN

; EXTERNAL INTERRUPT #0
      ORG      0003H
      AJMP     INT_0

; TIMER FLAG 0
      ORG      000BH
      RETI

; EXTERNAL INTERRUPT #1
      ORG      0013H
      RETI

; IIC TIMER OVERFLOW
      ORG      001BH
      RETI

; SERIAL PORT INTERRUPT
      ORG      0023H
      RETI

; ADC EOC
      ORG      002BH
      RETI

; PWM COUNTER OVERFLOW
      ORG      0033H
      RETI


;****************************************
;       DATA TABLE
;****************************************
;
      ORG      0040H

M1X1:
      DW       08800H

M2X2:
      DW       00BCDH, 00BCDH
      DW       08433H, 00433H

M3X3:
      DW       00D11H, 00D11H, 00D11H
      DW       00800H, 00800H, 00800H
      DW       082EFH, 002EFH, 002EFH

M4X4:
      DW       00DB3H, 00DB3H, 00DB3H, 00DB3H
      DW       009E6H, 009E6H, 009E6H, 009E6H
      DW       0061AH, 0061AH, 0061AH, 0061AH
      DW       0824DH, 0024DH, 0024DH, 0024DH

T_LENGTH:
      INC      A
      MOVC     A, @A+PC
      RET
      DB       02H, 08H, 012H, 020H


TABLE   EQU     020H

;************************************************************
;
;       MAIN PROGRAM
;
;       R7 CONTIENT L'OFFSET.
;
;       DPTR POINTE AU DEBUT DE LA TABLE COURANTE (ROM).
;
;************************************************************
;
      ORG      100H

MAIN:

      CLR      P0.3              ; CLEAR LATCH ENABLE ON LSB CHIP
      CLR      P0.4              ; CLEAR LATCH ENABLE ON MSB CHIP

      SETB     P1.7              ; CXOK/ = 1

      MOV      ADCON, #0E4H      ; ENABLE ADC CHANNEL #4

;
; DETERMINE THE MICROSCANNING MODE
;
      MOV      A, P0             ; READ DIP SWITCH
      ANL      A, #00000011B     ; APPLY MODE MASK


      CJNE     A, #00H, M2       ; SET DPTR TO THE BEGINNING OF
      MOV      DPTR, #M1X1       ; CORRESPONDING TABLE
      SJMP     DPTR_OK
M2:   CJNE     A, #01H, M3
      MOV      DPTR, #M2X2
      SJMP     DPTR_OK
M3:   CJNE     A, #02H, M4
      MOV      DPTR, #M3X3
      SJMP     DPTR_OK
M4:   MOV      DPTR, #M4X4
DPTR_OK:

;
; TRANSFER COMMAND TABLE IN RAM
;
      ACALL    T_LENGTH          ; DETERMINE THE SIZE OF THE TABLE
      MOV      R2, A             ; SAVE TABLE LENGTH
      MOV      R7, #00H          ; SET OFFSET TO 0
      MOV      R0, #TABLE
L1:   MOV      A, R7
      MOVC     A, @A+DPTR
      MOV      @R0, A
      INC      R0
      INC      R7
      DJNZ     R2, L1


      MOV      R7, #00H          ; INITIALIZE OFFSET TO 00H


;
; WRITE FIRST SAMPLE TO OUTPUT PORT
;
      ACALL    READ_RAM_TABLE    ; READ FIRST SAMPLE FROM TABLE
      ACALL    WRITE_PORT        ; WRITE SAMPLE TO OUTPUT PORT

      MOV      IE, #10000001B    ; ENABLE INTERRUPTS

LOOP:
      AJMP     LOOP

;************************************************************
;
; WRITE_PORT SUBROUTINE
;       INPUT:  R2 R3
;
;************************************************************
WRITE_PORT:
      MOV      P3, R2            ; WRITE MSB
      SETB     P0.4
      CLR      P0.4
      MOV      P3, R3            ; WRITE LSB
      SETB     P0.3
      CLR      P0.3
      RET


;************************************************************
;
; READ_RAM_TABLE SUBROUTINE
;       INPUT:  R7
;       OUTPUT: R2 R3
;       ACC AND R0 ARE MODIFIED
;
;************************************************************
READ_RAM_TABLE:
      MOV      A, #TABLE
      ADD      A, R7
      MOV      R0, A
      MOV      A, @R0
      MOV      R2, A
      INC      R0
      MOV      A, @R0
      MOV      R3, A
      RET


;************************************************************
;
; WRITE_RAM_TABLE SUBROUTINE
;       INPUT:  R7 R2 R3
;
;       ACC AND R0 ARE MODIFIED
;
;************************************************************
WRITE_RAM_TABLE:
      MOV      A, #TABLE
      ADD      A, R7
      MOV      R0, A
      MOV      A, R2
```

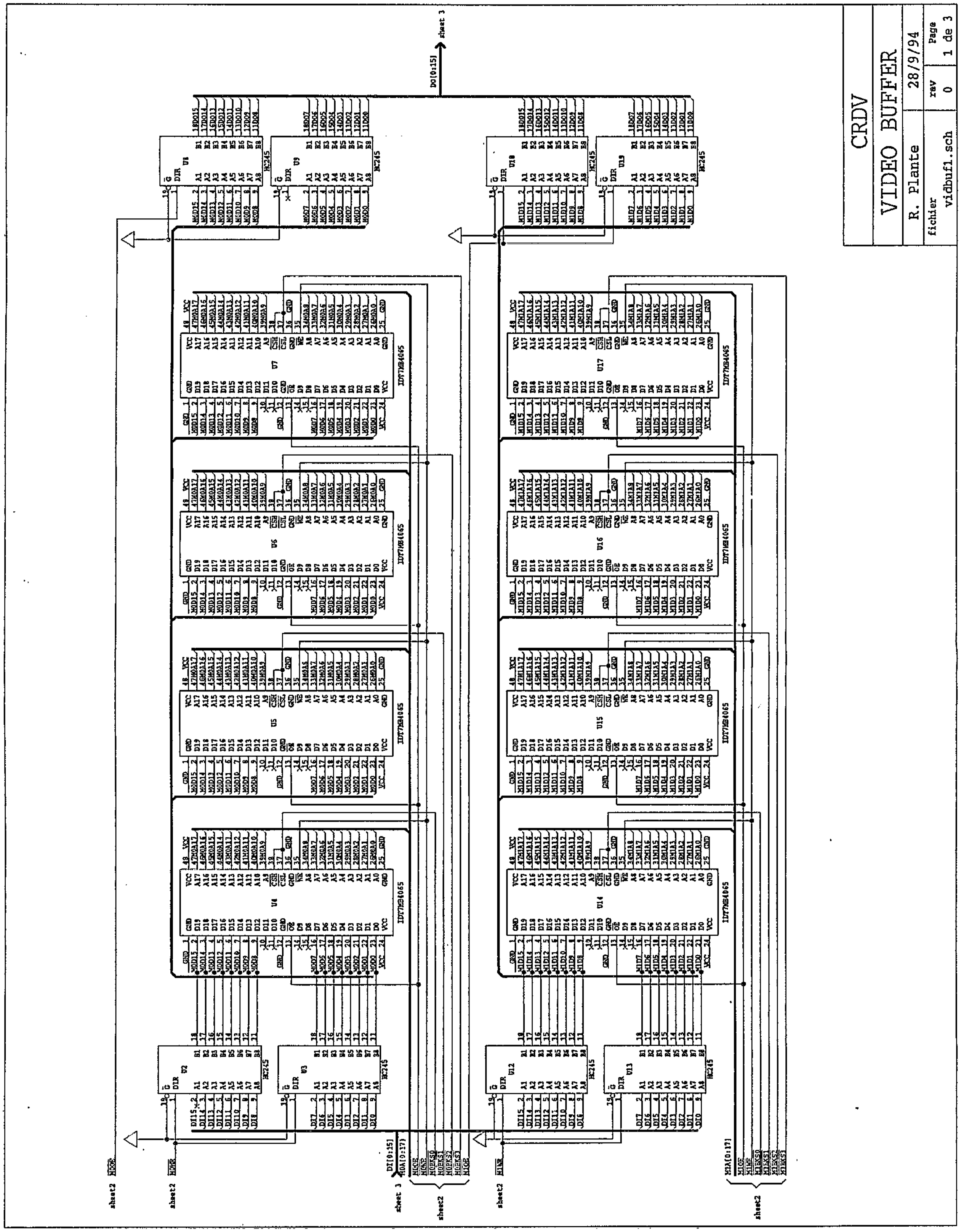
```
        MOV     @R0, A
        INC     R0
        MOV     A, R3
        MOV     @R0, A
        RET

;************************************************************
;
; READ_ROM_TABLE SUBROUTINE
;       INPUT:  R7, DPTR
;       OUTPUT: R4 R5
;
;       ACC IS MODIFIED
;
;************************************************************
READ_ROM_TABLE:
        MOV     A, R7
        MOVC    A, @A+DPTR
        MOV     R4, A
        MOV     A, R7
        INC     A
        MOVC    A, @A+DPTR
        MOV     R5, A
        RET

;************************************************************
;
; INT0 SUBROUTINE (LEVEL TRIGGERED)
;       ACC, R0, R2, R3 ARE MODIFIED
;
;************************************************************

INT_0:

;
; GO TO NEXT MICROSCANNING STEP AS FAST AS POSSIBLE
;

W1:     JB      P1.6, W1        ; SYNCHRONIZE CHANNEL Y ON X

        MOV     A, P1           ; GET THE NEW OFFSET VALUE FROM PORT
1
        ANL     A, #0FH
        RL      A               ; MULTIPLY BY 2
        MOV     R7, A           ; SAVE OFFSET VALUE

        CLR     P1.7            ; CHANNEL Y OK
W3:     JNB     P1.6, W3        ; WAIT FOR ACKNOWLEDGE
        SETB    P1.7

        ACALL   READ_RAM_TABLE
        ACALL   WRITE_PORT

W2:     JNB     P1.5, W2        ; WAIT UNTIL PIN 20 GOES HIGH

;
; THE CALIBRATION ROUTINE IS EXECUTED DURING THE INTEGRATION TIME.
; A LOT OF TIME IS THUS AVAILABLE.
;

        ACALL   SAMPLE          ; READ CURRENT POSITION (R2, R3)

        ACALL   READ_ROM_TABLE  ; GET IDEAL POSITION (R4, R5) AND
        MOV     A, R4           ; CLEAR INDEX BIT OF R4
        ANL     A, #0FH
        MOV     R4, A

        CLR     C               ; COMPUTE ERROR (R4R5 = R4R5-R2R3)
        MOV     A, R5
        SUBB    A, R3
        MOV     R5, A
        MOV     A, R4
        SUBB    A, R2
        MOV     R4, A

        ACALL   READ_RAM_TABLE  ; ADJUST COMMAND IN RAM TABLE
        MOV     A, R2           ; SAVE TABLE INDEX FOR FUTURE USE
        ANL     A, #0F0H
        PUSH    ACC
        MOV     A, R2           ; CLEAR INDEX BITS OF R2
        ANL     A, #0FH
        MOV     R2, A

        MOV     A, R3           ; ADD ERROR (R2R3 = R2R3 + R4R5)
        ADD     A, R5
        MOV     R3,A
        MOV     A, R2
        ADDC    A, R4
        MOV     R2, A

        MOV     A, R2           ; CHECK R2R3 FOR OUT OF RANGE VALUE
        ANL     A, #0F0H
        JZ      RANGE_OK
        ANL     A, #080H
        JZ      OVER
        MOV     R2, #00H
        MOV     R3, #00H
        AJMP    RANGE_OK
OVER:   MOV     R2, #0FH
        MOV     R3, #0FFH

RANGE_OK:
        POP     ACC             ; RESTORE INDEX BITS IN R2
        ORL     A, R2
        MOV     R2, A

        ACALL   WRITE_RAM_TABLE ; SAVE MODIFIED VALUE IN RAM TABLE

INT0_END:
        RETI

;************************************************************
;
; SAMPLING ROUTINE
;       THE VALUE IS RETURNED IN R2(MSB) AND R3(LSB)
;       ACC AND R4 ARE MODIFIED.
;
;************************************************************

SAMPLE:
        MOV     R2, #00H        ; INITIALIZE RESULT
        MOV     R3, #00H

        MOV     R4, #010H       ; SUM 16 SAMPLES

        MOV     A, ADAT         ; CLEAR ADCI

START_ADC:
        MOV     A, ADCON        ; START CONVERSION
        ORL     A, #08H
        MOV     ADCON, A

WAIT_EOC:                       ; WAIT UNTIL END OF CONVERSION
        MOV     A, ADCON
        ANL     A, #010H
        JZ      WAIT_EOC

        MOV     A, ADAT         ; READ RESULT

        ADD     A, R3           ; ADD RESULT
        MOV     R3, A
        CLR     A
        ADDC    A, R2
        MOV     R2, A

        DJNZ    R4, START_ADC   ; LOOP R4 TIMES

        RET

        END
```

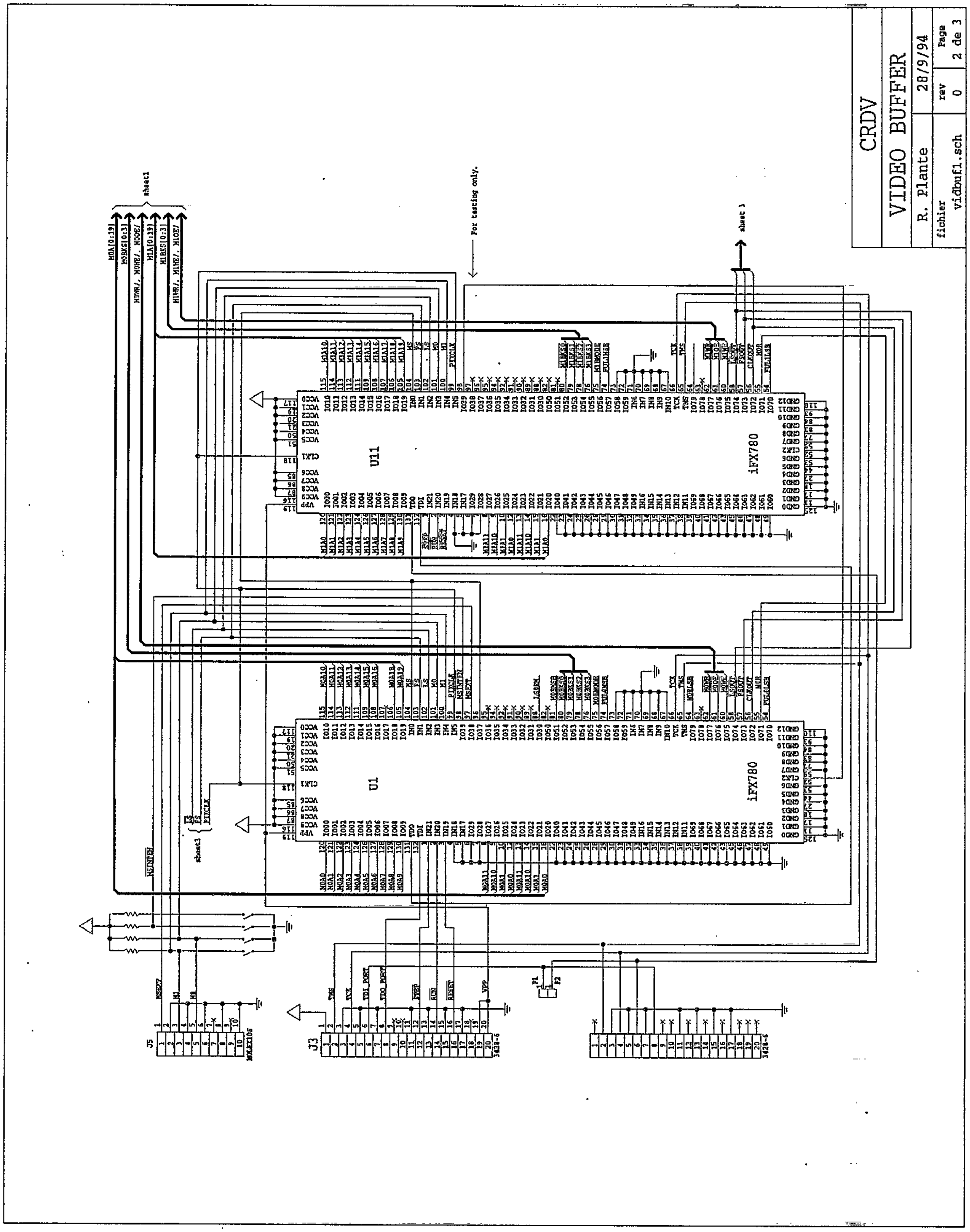
APPENDIX C

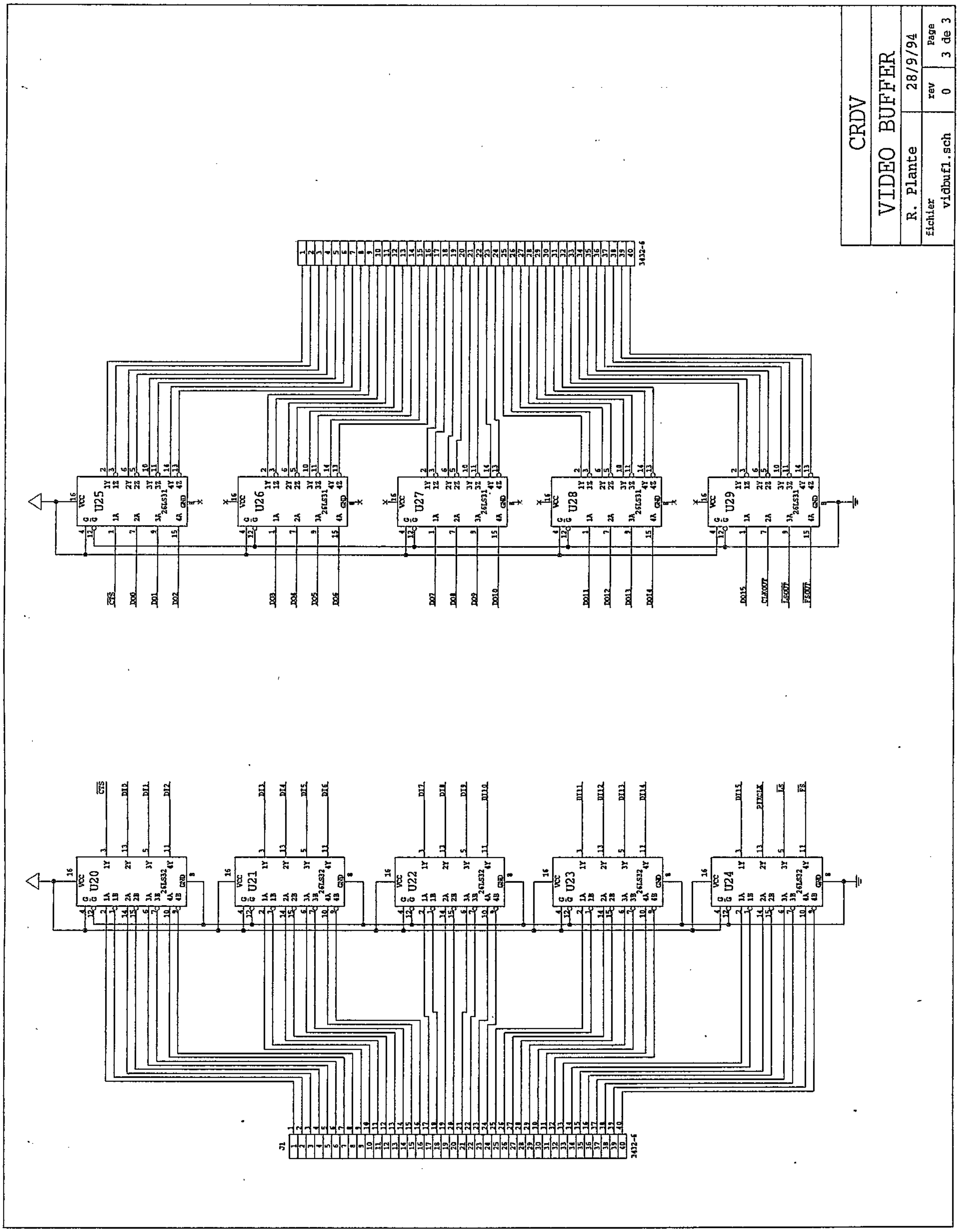
Position Sensor Conditioning Amplifier



APPENDIX D
Real Time Microscanning Processor


CRDV
VIDEO BUFFER
R. Plante
28/9/94
fichier
vidbuf1.sch
rev
0
Page
1 de 3


CRDV
VIDEO BUFFER
R. Plante
28/9/94
fichier
vidbuf1.sch
rev
0
Page
2 de 3
For testing only.
sheet 1
sheet 3
U11
iFX780
U1
iFX780
J5
J3


CRDV
VIDEO BUFFER
R. Plante
28/9/94
fichier
vidbuf1.sch
rev
0
Page
3 de 3

```
TITLE    video buffer
PATTERN
REVISION A
AUTHOR   Robert Plante
COMPANY  CRDV
DATE     09/07/96

CHIP     VIDU1_H iFX780_132

; Project Infra-red Eye
;

; This is the source file for one of the iFX780 on the Video Buffer (U1)
; The video buffer takes n (n=1,4,16) consecutive micro-scannned images
; of 256x256 pixels, store them in a proper sequence into a dual-buffer.
; When filled, the buffer is swapped with the other one to be fill.
; The first buffer is then scanned in a linear fashion to get a full
; oversampled image.

; This FPGA is used to generated the address and timings for the first
; buffer. Three modes of microscanning are available: 1x1, 2x2, 4x4.


; version 1: first operational version
; version 2: modified to correct for sequential scan integration
;
;--------------------------------- PIN Declarations --------------


PIN   118   /PIXCLK      TTL_LEVEL

;Memory Buffer counters for address
PIN   IO00  MBA0
PIN   IO01  MBA1
PIN   IO02  MBA2
PIN   IO03  MBA3
PIN   IO04  MBA4
PIN   IO05  MBA5
PIN   IO06  MBA6
PIN   IO07  MBA7
PIN   IO08  MBA8
PIN   IO09  MBA9
PIN   IO10  MBA10
PIN   IO11  MBA11
PIN   IO12  MBA12
PIN   IO13  MBA13
PIN   IO14  MBA14
PIN   IO15  MBA15
PIN   IO16  MBA16
PIN   IO17  MBA17
PIN   IO18  MBA18
PIN   IO19  MBA19

;Memory Buffer counters for address, write mode 2x2, 4x4

PIN   IO20  MM4A0       ; this output is used instead of MBA0
PIN   IO21  MM4A1       ; this output is used instead of MBA1
PIN   IO22  MM4A10      ; this output is used instead of MBA10
PIN   IO23  MM2A11      ; this output is used instead of MBA11
PIN   IO24  MM2A0       ; this output is used instead of MBA0
PIN   IO25  MM2A1       ; not used
PIN   IO26  MM2A10      ; this output is used instead of MBA0
PIN   IO27  MM4A11      ; not used

PIN   IO30  LSEN  ; Intermediate value which enable counting LS pulses
PIN   IO31  MWET  ; Intermediate value to create a delay for MWE
PIN   IO32  BINA512     ; Intermediate value because of lack of Pterm

; Frame sync counter for generating MS sync
PIN   IO33  FSCT0
PIN   IO34  FSCT1
PIN   IO35  FSCT2
PIN   IO36  FSCT3
PIN   IO37  /MSINT      ; MicroScan output

; Spare pins, no connect -> program as output
PIN   IO38  MSTRG       ; External MS, must be AND with FS
PIN   IO39  /MSINTEN    ; Enable internal MS generation
PIN   IO50  NC3

PIN   IO51  BRESMSB     ; Reset binary counter MSB
PIN   IO52  /BKS0       ; BANK SELECT 0
PIN   IO53  /BKS1       ; BANK SELECT 1
PIN   IO54  /BKS2       ; BANK SELECT 2
PIN   IO55  /BKS3       ; BANK SELECT 3
PIN   IO56  BINMODE     ; Binary mode
PIN   IO57  FULLMSB     ; Memory msb counter full


PIN   IO70  FULLLSB     ; Memory lsb counter full
PIN   IO71  MOR         ; Memory in read mode
PIN   IO72  CLKOUT      ; Pixel clock out
PIN   IO73  /FSOUT      ; Frame sync out
PIN   IO74  /LSOUT      ; Line sync out
PIN   IO75  /MWE        ; Memory buffer WE/
PIN   IO76  /MOE        ; Memory buffer OE/
PIN   IO77  /MWR        ; Memory transceiver enable for write
PIN   IO78  ENCNT       ; Enable counting,may be an internal signal
PIN   IO79  BRESLSB     ; Reset binary counter LSB


PIN   IN0   /MS         ; Micro scanning sync pulse
```

```
PIN   IN1   /FS       ; Frame sync pulse
PIN   IN2   /LS       ; Line sync pulse
PIN   IN3   M0        ; Mode bit0
PIN   IN4   M1        ; Mode bit1
PIN   IN5   PIXCLK2   TTL_LEVEL

STRING MOW  ' (/MOR) '
STRING MODE1X1    ' (/M1 * /M0) '
STRING MODE2X2    ' (/M1 * M0) '
STRING MODE4X4    ' ( M1 * M0) '
STRING MODE512    ' (MODE2X2 * MOW) '
STRING MODE1024   ' (MODE4X4 * MOW) '


STATE
MOORE_MACHINE
DEFAULT_BRANCH HOLD_STATE

Z0 = /FSCT3 * /FSCT2 * /FSCT1 * /FSCT0
Z1 = /FSCT3 * /FSCT2 * /FSCT1 * FSCT0
Z2 = /FSCT3 * /FSCT2 * FSCT1 * /FSCT0
Z3 = /FSCT3 * /FSCT2 * FSCT1 * FSCT0
Z4 = /FSCT3 * FSCT2 * /FSCT1 * /FSCT0
Z5 = /FSCT3 * FSCT2 * /FSCT1 * FSCT0
Z6 = /FSCT3 * FSCT2 * FSCT1 * /FSCT0
Z7 = /FSCT3 * FSCT2 * FSCT1 * FSCT0
Z8 = FSCT3 * /FSCT2 * /FSCT1 * /FSCT0
Z9 = FSCT3 * /FSCT2 * /FSCT1 * FSCT0
Z10 = FSCT3 * /FSCT2 *  FSCT1 * /FSCT0
Z11 = FSCT3 * /FSCT2 *  FSCT1 * FSCT0
Z12 = FSCT3 * FSCT2 * /FSCT1 * /FSCT0
Z13 = FSCT3 * FSCT2 * /FSCT1 * FSCT0
Z14 = FSCT3 * FSCT2 * FSCT1 * /FSCT0
Z15 = FSCT3 * FSCT2 * FSCT1 * FSCT0

Z0 := FS -> Z1
Z1 := FS -> Z2
Z2 := FS -> Z3
Z3 := FS -> Z4
Z4 := FS -> Z5
Z5 := FS -> Z6
Z6 := FS -> Z7
Z7 := FS -> Z8
Z8 := FS -> Z9
Z9 := FS -> Z10
Z10:= FS -> Z11
Z11:= FS -> Z12
Z12:= FS -> Z13
Z13:= FS -> Z14
Z14:= FS -> Z15
Z15:= FS -> Z0

STATE
MOORE_MACHINE
DEFAULT_BRANCH HOLD_STATE

;S0 = /MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
;S1 = /MM4A11 * /MM4A10 * /MM4A1 * MM4A0
;S2 = /MM4A11 * /MM4A10 *MM4A1 * /MM4A0
;S3 = /MM4A11 * /MM4A10 *MM4A1*MM4A0
;S4 = /MM4A11 * MM4A10 * /MM4A1 * /MM4A0
;S5 = /MM4A11 * MM4A10 * /MM4A1 * MM4A0
;S6 = /MM4A11 * MM4A10 * MM4A1 * /MM4A0
;S7 = /MM4A11 * MM4A10 * MM4A1 * MM4A0
;S8 = MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
;S9 = MM4A11 * /MM4A10 * /MM4A1 * MM4A0
;S10 = MM4A11 * /MM4A10 * MM4A1 * /MM4A0
;S11 = MM4A11 * /MM4A10 * MM4A1 * MM4A0
;S12 = MM4A11 * MM4A10 * /MM4A1 * /MM4A0
;S13 = MM4A11 * MM4A10 * /MM4A1 * MM4A0
;S14 = MM4A11 * MM4A10 * MM4A1 * /MM4A0
;S15 = MM4A11 * MM4A10 * MM4A1 * MM4A0

S0 = /MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
S1 = /MM4A11 * /MM4A10 * /MM4A1 * MM4A0
S2 = /MM4A11 * /MM4A10 * MM4A1 * /MM4A0
S3 = /MM4A11 * /MM4A10 * MM4A1 * MM4A0
S4 = /MM4A11 * MM4A10 * MM4A1 * MM4A0
S5 = /MM4A11 * MM4A10 * MM4A1 * /MM4A0
S6 = MM4A11 * /MM4A10 * MM4A1 * /MM4A0
S7 = MM4A11 * /MM4A10 * MM4A1 * MM4A0
S8 = MM4A11 * MM4A10 * MM4A1 * MM4A0
S9 = MM4A11 * MM4A10 * MM4A1 * /MM4A0
S10 = MM4A11 * MM4A10 * /MM4A1 * MM4A0
S11 = MM4A11 * MM4A10 * /MM4A1 * /MM4A0
S12 = MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
S13 = MM4A11 * /MM4A10 * /MM4A1 * MM4A0
S14 = /MM4A11 * MM4A10 * /MM4A1 * MM4A0
S15 = /MM4A11 * MM4A10 * /MM4A1 * /MM4A0


S0 := FS*/MS -> S1
 + MS -> S0
S1 := FS*/MS -> S2
 + MS -> S0
S2 := FS*/MS -> S3
 + MS -> S0
S3 := FS*/MS -> S4
 + MS -> S0
S4 := FS*/MS -> S5
 + MS -> S0
```

```
S5 := FS*/MS -> S6
 + MS -> S0
S6 := FS*/MS -> S7
 + MS -> S0
S7 := FS*/MS -> S8
 + MS -> S0
S8 := FS*/MS -> S9
 + MS -> S0
S9 := FS*/MS -> S10
 + MS -> S0
S10:= FS*/MS -> S11
 + MS -> S0
S11:= FS*/MS -> S12
 + MS -> S0
S12:= FS*/MS -> S13
 + MS -> S0
S13:= FS*/MS -> S14
 + MS -> S0
S14:= FS*/MS -> S15
 + MS -> S0
S15:= MS -> S0


STATE
MOORE_MACHINE
DEFAULT_BRANCH HOLD_STATE

X0 = /MM2A10 * /MM2A0
X1 = /MM2A10 * MM2A0
X2 = MM2A10 * MM2A0
X3 = MM2A10 * /MM2A0

X0 := FS*/MS -> X1
 + MS -> X0
X1 := FS*/MS -> X2
 + MS -> X0
X2 := FS*/MS -> X3
 + MS -> X0
X3 := MS -> X0


EQUATIONS

BINA512 = BINMODE + MODE512
NC3 = GND
MOR.D := /MOR*MS + MOR*/MS
MOE = MOR
MWR = /MOR
MWET = /MOR * /LS * PIXCLK2 * /(FULLLSB * FULLMSB)
MWE = MWET
MSINT = MSINTEN * (FS*(MODE1X1 + MODE2X2*FSCT1*FSCT0 +
      MODE4X4*FSCT3*FSCT2*FSCT1*FSCT0)) +
      /MSINTEN * (MSTRG * FS)
BINMODE = (MODE1X1 + MODE2X2 * MOR + MODE4X4 * MOR)
BRESLSB.D := MOR*LS*(MODE1X1 + MODE2X2*MBA8
  + MODE4X4 * MBA7*MBA8*MBA9) + MOW*LS
BRESMSB.D := (BINMODE*MS + MODE512*FS + MODE1024*FS)
FULLLSB = (MODE1X1 * MBA0*MBA1*MBA2*MBA3*MBA4*MBA5*MBA6*MBA7 +
  MODE2X2 * MBA0*MBA1*MBA2*MBA3*MBA4*MBA5*MBA6*MBA7*MBA8 +
MODE4X4 * MBA0*MBA1*MBA2*MBA3*MBA4*MBA5*MBA6*MBA7* MBA8* MBA9) * BINMODE +
  (MODE512 * MBA1*MBA2*MBA3*MBA4*MBA5*MBA6*MBA7* MBA8) +
  (MODE1024 * MBA2*MBA3*MBA4*MBA5*MBA6*MBA7* MBA8* MBA9)
FULLMSB = (MODE1X1 * MBA10*MBA11*MBA12*MBA13*MBA14*MBA15*MBA16*MBA17 +
  MODE2X2 * MBA10*MBA11*MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18 +
  MODE4X4 * MBA10*MBA11*MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18* MBA19)
* BINMODE +
  (MODE512 * MBA11*MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18*/MBA19) +
  (MODE1024 * MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18* MBA19)

BKS0 = /MBA19 * /MBA18
BKS1 = /MBA19 * MBA18
BKS2 = MBA19 * /MBA18
BKS3 = MBA19 * MBA18

MBA0.D := ENCNT*/FULLLSB*/MBA0 + MBA0*(/ENCNT+FULLLSB)
MBA1.T := BINMODE*ENCNT*/FULLLSB*(MBA0)+
  MODE512*ENCNT*/FULLLSB
MBA2.T := BINMODE*ENCNT*/FULLLSB*(MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA1)+
  MODE1024*ENCNT*/FULLLSB
MBA3.T := BINMODE*ENCNT*/FULLLSB*(MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA2)
MBA4.T := BINMODE*ENCNT*/FULLLSB*(MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA3*MBA2)
MBA5.T := BINMODE*ENCNT*/FULLLSB*(MBA4*MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA4*MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA4*MBA3*MBA2)
MBA6.T := BINMODE*ENCNT*/FULLLSB*(MBA5*MBA4*MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA5*MBA4*MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA5*MBA4*MBA3*MBA2)
MBA7.T := BINMODE*ENCNT*/FULLLSB*(MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA6*MBA5*MBA4*MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA6*MBA5*MBA4*MBA3*MBA2)
MBA8.T := BINMODE*ENCNT*/FULLLSB*(MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA7*MBA6*MBA5*MBA4*MBA3*MBA2)
MBA9.T:=
  BINMODE*ENCNT*/FULLLSB*(MBA8*MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA8*MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA8*MBA7*MBA6*MBA5*MBA4*MBA3*MBA2)
```

```
LSEN = (MODE1X1+MODE2X2*MBA8+MODE4X4*MBA7*MBA8*MBA9)

MBA10.T := LS*ENCNT*/FULLMSB*LSEN
MBA11.T := LS*ENCNT*/FULLMSB*(MBA10*BINMODE*LSEN + MODE512)
MBA12.T := LS*ENCNT*/FULLMSB*
  (MBA10*MBA11*BINMODE*LSEN+MBA11*MODE512+ MODE1024)
MBA13.T := LS*BINMODE*LSEN*ENCNT*/FULLMSB*(MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA12)
MBA14.T := LS*BINMODE*LSEN*ENCNT*/FULLMSB*(MBA13*MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA13*MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA13*MBA12)
MBA15.T := LS*BINMODE*LSEN*ENCNT*/FULLMSB*(MBA14*MBA13*MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA14*MBA13*MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA14*MBA13*MBA12)
MBA16.T :=
  LS*BINMODE*LSEN*ENCNT*/FULLMSB*(MBA15*MBA14*MBA13*MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA15*MBA14*MBA13*MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA15*MBA14*MBA13*MBA12)
MBA17.T :=
  LS*BINMODE*LSEN*ENCNT*/FULLMSB*
  (MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA16*MBA15*MBA14*MBA13*MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA16*MBA15*MBA14*MBA13*MBA12)
MBA18.T :=
  LS*BINMODE*LSEN*ENCNT*/FULLMSB*
  (MBA17*MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA17*MBA16*MBA15*MBA14*MBA13*MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA17*MBA16*MBA15*MBA14*MBA13*MBA12)
MBA19.T :=
  LS*BINMODE*LSEN*ENCNT*/FULLMSB*
  (MBA18*MBA17*MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*MBA10)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA18*MBA17*MBA16*MBA15*MBA14*MBA13*MBA12)

BRESLSB.CLKF = PIXCLK
BRESMSB.CLKF = PIXCLK
MOR.CLKF = PIXCLK
MBA0.CLKF = PIXCLK
MBA1.CLKF = PIXCLK
MBA2.CLKF = PIXCLK
MBA3.CLKF = PIXCLK
MBA4.CLKF = PIXCLK
MBA5.CLKF = PIXCLK
MBA6.CLKF = PIXCLK
MBA7.CLKF = PIXCLK
MBA8.CLKF = PIXCLK
MBA9.CLKF = PIXCLK
MBA10.CLKF = PIXCLK
MBA11.CLKF = PIXCLK
MBA12.CLKF = PIXCLK
MBA13.CLKF = PIXCLK
MBA14.CLKF = PIXCLK
MBA15.CLKF = PIXCLK
MBA16.CLKF = PIXCLK
MBA17.CLKF = PIXCLK
MBA18.CLKF = PIXCLK
MBA19.CLKF = PIXCLK

MM4A0.CLKF = PIXCLK
MM4A1.CLKF = PIXCLK
MM4A10.CLKF = PIXCLK
MM4A11.CLKF = PIXCLK

MM2A0.CLKF = PIXCLK
MM2A1.CLKF = PIXCLK
MM2A10.CLKF = PIXCLK
MM2A11.CLKF = PIXCLK

FSCT0.CLKF = /PIXCLK
FSCT1.CLKF = /PIXCLK
FSCT2.CLKF = /PIXCLK
FSCT3.CLKF = /PIXCLK

FSCT0.TRST = VCC
FSCT1.TRST = VCC
FSCT2.TRST = VCC
FSCT3.TRST = VCC

BRESLSB.TRST = VCC
BRESMSB.TRST = VCC
MOR.TRST = VCC
MSINT.TRST = VCC

MBA0.TRST = BINMODE
MBA1.TRST = BINA512
MBA2.TRST = VCC
MBA3.TRST = VCC
MBA4.TRST = VCC
MBA5.TRST = VCC
MBA6.TRST = VCC
MBA7.TRST = VCC
MBA8.TRST = VCC
MBA9.TRST = VCC
MBA10.TRST = BINMODE
MBA11.TRST = BINA512
MBA12.TRST = VCC
MBA13.TRST = VCC
MBA14.TRST = VCC
MBA15.TRST = VCC
MBA16.TRST = VCC
MBA17.TRST = VCC
```

```
MBA18.TRST = VCC
MBA19.TRST = VCC


MBA0.RSTF = BRESLSB
MBA1.RSTF = BRESLSB
MBA2.RSTF = BRESLSB
MBA3.RSTF = BRESLSB
MBA4.RSTF = BRESLSB
MBA5.RSTF = BRESLSB
MBA6.RSTF = BRESLSB
MBA7.RSTF = BRESLSB
MBA8.RSTF = BRESLSB
MBA9.RSTF = BRESLSB
MBA10.RSTF = BRESMSB
MBA11.RSTF = BRESMSB
MBA12.RSTF = BRESMSB
MBA13.RSTF = BRESMSB
MBA14.RSTF = BRESMSB
MBA15.RSTF = BRESMSB
MBA16.RSTF = BRESMSB
MBA17.RSTF = BRESMSB
MBA18.RSTF = BRESMSB
MBA19.RSTF = BRESMSB

MM4A0.TRST = MODE1024
MM4A1.TRST = MODE1024
MM4A10.TRST = MODE1024
MM4A11.TRST = MODE1024

MM2A0.TRST = MODE512
MM2A1.TRST = GND
MM2A10.TRST = MODE512
MM2A11.TRST = GND

MM2A1.D := /LS*/FULLLSB*/MM2A1 + (LS+/ENCNT+FULLLSB)*MM2A1*/BRESLSB
MM2A11.D := LS*/FULLMSB*/MM2A11 + (/LS+/ENCNT+FULLMSB)*MM2A11*/BRESMSB


CLKOUT = PIXCLK2
LSOUT.D := LS*(MODE1X1 + MODE2X2*MBA8 + MODE4X4*MBA9*MBA8*MBA7)
FSOUT.D := MS
LSOUT.CLKF = PIXCLK
FSOUT.CLKF = PIXCLK
LSOUT.TRST = MOR
FSOUT.TRST = MOR

SIMULATION

VECTOR COUNT := [MBA8, MBA7, MBA6, MBA5, MBA4, MBA3, MBA2, MBA1, MBA0]


TRACE_ON BRESLSB BRESMSB BKS0 BKS1 BKS2 BKS3 BINMODE FULLMSB FULLLSB
  MOR CLKOUT FSOUT LSOUT MWE MOE MWR PIXCLK
  MBA8 MBA7 MBA6 MBA5 MBA4 MBA3 MBA2 MBA1 MBA0

SETF ENCNT /MS /FS /LS M0 /M1 PIXCLK /PIXCLK2

PRLDF MOR /BRESLSB /BRESMSB /MBA9 MBA8 MBA7 /MBA6 /MBA5 /MBA4 /MBA3 /MBA2
PRLDF /MBA1 /MBA0 /MBA19 /MBA18 /MBA17 /MBA16 /MBA15 /MBA14 /MBA13 /MBA12
/MBA11 /MBA10

FOR X:=0 TO 127 DO
 BEGIN
  CLOCKF PIXCLK
 END
SETF LS
CLOCKF PIXCLK
SETF /LS
FOR X:=0 TO 10 DO
 BEGIN
  CLOCKF PIXCLK
 END

TRACE_OFF
```

```
TITLE video buffer
PATTERN
REVISION A
AUTHOR Robert Plante
COMPANY CRDV
DATE 09/07/96

CHIP VIDU11_H iFX780_132

; Project Infra-red Eye
;

; This is the source file for one of the iFX780 on the Video Buffer (U11)
; The video buffer takes n (n=1,4,16) consecutive micro-scannned images
; of 256x256 pixels, store them in a proper sequence into a dual-buffer.
; When filled, the buffer is swapped with the other one to be fill.
; The first buffer is then scanned in a linear fashion to get a full
; oversampled image.

; This FPGA is used to generated the address and timings for the second
; buffer. Three modes of microscanning are available: 1x1, 2x2, 4x4.

; version 1: first operational version
; version 2: modified to correct for sequential scan integration
;
;--------------------------------- PIN Declarations --------------


PIN 118 /PIXCLK TTL_LEVEL

;Memory Buffer counters for address
PIN IO00 MBA0
PIN IO01 MBA1
PIN IO02 MBA2
PIN IO03 MBA3
PIN IO04 MBA4
PIN IO05 MBA5
PIN IO06 MBA6
PIN IO07 MBA7
PIN IO08 MBA8
PIN IO09 MBA9
PIN IO10 MBA10
PIN IO11 MBA11
PIN IO12 MBA12
PIN IO13 MBA13
PIN IO14 MBA14
PIN IO15 MBA15
PIN IO16 MBA16
PIN IO17 MBA17
PIN IO18 MBA18
PIN IO19 MBA19

;Memory Buffer counters for address, write mode 2x2, 4x4

PIN IO20 MM4A0     ; this output is used instead of MBA0
PIN IO21 MM4A1     ; this output is used instead of MBA1
PIN IO22 MM4A10    ; this output is used instead of MBA10
PIN IO23 MM2A11    ; not used
PIN IO24 MM2A0     ; this output is used instead of MBA0
PIN IO25 MM2A1     ; not used
PIN IO26 MM2A10    ; this output is used instead of MBA0
PIN IO27 MM4A11    ; this output is used instead of MBA11

PIN IO30 LSEN      ; Intermediate value which enable counting LS pulses
PIN IO31 MWET      ; Intermediate value to create a delay for MWE
PIN IO32 BINA512   ; Intermediate value because of lack of Pterm

; Spare pins, no connect -> program as output
PIN IO33 NC1
PIN IO34 NC2
PIN IO35 NC3
PIN IO36 NC4
PIN IO37 NC5
PIN IO38 NC6
PIN IO39 NC7
PIN IO50 NC8


PIN IO51 BRESMSB   ; Reset binary counter MSB
PIN IO52 /BKS0     ; BANK SELECT 0
PIN IO53 /BKS1     ; BANK SELECT 1
PIN IO54 /BKS2     ; BANK SELECT 2
PIN IO55 /BKS3     ; BANK SELECT 3
PIN IO56 BINMODE   ; Binary mode
PIN IO57 FULLMSB   ; Memory msb counter full
PIN IO71 M0R       ; Memory-0 in read mode

; Output signals

PIN IO70 FULLLSB   ; Memory lsb counter full
PIN IO72 CLKOUT    ; Pixel clock out
PIN IO73 /FSOUT    ; Frame sync out
PIN IO74 /LSOUT    ; Line sync out
PIN IO75 /MWE      ; Memory buffer WE/
PIN IO76 /MOE      ; Memory buffer OE/
PIN IO77 /MWR      ; Memory transceiver enable for write
PIN IO78 ENCNT     ; Enable counting,may be an internal signal
PIN IO79 BRESLSB   ; Reset binary counter LSB


PIN IN0  /MS       ; Micro scanning sync pulse
PIN IN1  /FS       ; Frame sync pulse
PIN IN2  /LS       ; Line sync pulse
PIN IN3  M0        ; Mode bit0
PIN IN4  M1        ; Mode bit1
PIN IN5  PIXCLK2 TTL_LEVEL


STRING M1R ' (/M0R) '
STRING M1W ' ( M0R) '
STRING MODE1X1 ' (/M1 * /M0) '
STRING MODE2X2 ' (/M1 * M0) '
STRING MODE4X4 ' ( M1 * M0) '
STRING MODE512 ' (MODE2X2 * M1W) '
STRING MODE1024 ' (MODE4X4 * M1W) '


STATE
MOORE_MACHINE
DEFAULT_BRANCH HOLD_STATE

;S0 = /MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
;S1 = /MM4A11 * /MM4A10 * /MM4A1 * MM4A0
;S2 = /MM4A11 * /MM4A10 * MM4A1 * /MM4A0
;S3 = /MM4A11 * /MM4A10 * MM4A1 * MM4A0
;S4 = /MM4A11 * MM4A10 * /MM4A1 * /MM4A0
;S5 = /MM4A11 * MM4A10 * /MM4A1 * MM4A0
;S6 = /MM4A11 * MM4A10 * MM4A1 * /MM4A0
;S7 = /MM4A11 * MM4A10 * MM4A1 * MM4A0
;S8 = MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
;S9 = MM4A11 * /MM4A10 * /MM4A1 * MM4A0
;S10 = MM4A11 * /MM4A10 * MM4A1 * /MM4A0
;S11 = MM4A11 * /MM4A10 * MM4A1 * MM4A0
;S12 = MM4A11 * MM4A10 * /MM4A1 * /MM4A0
;S13 = MM4A11 * MM4A10 * /MM4A1 * MM4A0
;S14 = MM4A11 * MM4A10 * MM4A1 * /MM4A0
;S15 = MM4A11 * MM4A10 * MM4A1 * MM4A0

S0 = /MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
S1 = /MM4A11 * /MM4A10 * /MM4A1 * MM4A0
S2 = /MM4A11 * /MM4A10 * MM4A1 * /MM4A0
S3 = /MM4A11 * /MM4A10 * MM4A1 * MM4A0
S4 = /MM4A11 * MM4A10 * MM4A1 * MM4A0
S5 = /MM4A11 * MM4A10 * MM4A1 * /MM4A0
S6 = MM4A11 * /MM4A10 * MM4A1 * /MM4A0
S7 = MM4A11 * /MM4A10 * MM4A1 * MM4A0
S8 = MM4A11 * MM4A10 * MM4A1 * MM4A0
S9 = MM4A11 * MM4A10 * MM4A1 * /MM4A0
S10 = MM4A11 * MM4A10 * /MM4A1 * MM4A0
S11 = MM4A11 * MM4A10 * /MM4A1 * /MM4A0
S12 = MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
S13 = MM4A11 * /MM4A10 * /MM4A1 * MM4A0
S14 = /MM4A11 * MM4A10 * /MM4A1 * MM4A0
S15 = /MM4A11 * MM4A10 * /MM4A1 * /MM4A0


S0 := FS*/MS -> S1
 + MS -> S0
S1 := FS*/MS -> S2
 + MS -> S0
S2 := FS*/MS -> S3
 + MS -> S0
S3 := FS*/MS -> S4
 + MS -> S0
S4 := FS*/MS -> S5
 + MS -> S0
S5 := FS*/MS -> S6
 + MS -> S0
S6 := FS*/MS -> S7
 + MS -> S0
S7 := FS*/MS -> S8
 + MS -> S0
S8 := FS*/MS -> S9
 + MS -> S0
S9 := FS*/MS -> S10
 + MS -> S0
S10:= FS*/MS -> S11
 + MS -> S0
S11:= FS*/MS -> S12
 + MS -> S0
S12:= FS*/MS -> S13
 + MS -> S0
S13:= FS*/MS -> S14
 + MS -> S0
S14:= FS*/MS -> S15
 + MS -> S0
S15:= MS -> S0

STATE
MOORE_MACHINE
DEFAULT_BRANCH HOLD_STATE


X0 = /MM2A10 * /MM2A0
X1 = /MM2A10 * MM2A0
X2 = MM2A10 * MM2A0
X3 = MM2A10 * /MM2A0

X0 := FS*/MS -> X1
 + MS -> X0
X1 := FS*/MS -> X2
 + MS -> X0
X2 := FS*/MS -> X3
```

```
 + MS -> X0
X3 := MS -> X0


EQUATIONS

BINA512 = BINMODE + MODE512
NC1 = GND
NC2 = GND
NC3 = GND
NC4 = GND
NC5 = GND
NC6 = GND
NC7 = GND
NC8 = GND

MOE = M1R

MWR = /M1R
MWET = /M1R * /LS * PIXCLK2 * /(FULLLSB * FULLMSB)
MWE = MWET
BINMODE = (MODE1X1 + MODE2X2 * M1R + MODE4X4 * M1R)
BRESLSB.D := M1R*LS*(MODE1X1 + MODE2X2*MBA8
  + MODE4X4 * MBA7*MBA8*MBA9) + M1W*LS
BRESMSB.D := (BINMODE*MS + MODE512*FS + MODE1024*FS)
FULLLSB = (MODE1X1 * MBA0*MBA1*MBA2*MBA3*MBA4*MBA5*MBA6*MBA7 +
  MODE2X2 * MBA0*MBA1*MBA2*MBA3*MBA4*MBA5*MBA6*MBA7*MBA8 +
  MODE4X4 * MBA0*MBA1*MBA2*MBA3*MBA4*MBA5*MBA6*MBA7*MBA8*MBA9)*BINMODE +
  (MODE512 * MBA1*MBA2*MBA3*MBA4*MBA5*MBA6*MBA7* MBA8) +
  (MODE1024 * MBA2*MBA3*MBA4*MBA5*MBA6*MBA7* MBA8* MBA9)
FULLMSB = (MODE1X1 * MBA10*MBA11*MBA12*MBA13*MBA14*MBA15*MBA16*MBA17 +
  MODE2X2 * MBA10*MBA11*MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18 +
  MODE4X4 * MBA10*MBA11*MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18* MBA19)
  *BINMODE +
  (MODE512 * MBA11*MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18*/MBA19) +
  (MODE1024 * MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18* MBA19)

BKS0 = /MBA19 * /MBA18
BKS1 = /MBA19 * MBA18
BKS2 = MBA19 * /MBA18
BKS3 = MBA19 * MBA18

MBA0.D := ENCNT*/FULLLSB*/MBA0 + MBA0*(/ENCNT+FULLLSB)
MBA1.T := BINMODE*ENCNT*/FULLLSB*(MBA0)+
  MODE512*ENCNT*/FULLLSB
MBA2.T := BINMODE*ENCNT*/FULLLSB*(MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA1)+
  MODE1024*ENCNT*/FULLLSB
MBA3.T := BINMODE*ENCNT*/FULLLSB*(MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA2)
MBA4.T := BINMODE*ENCNT*/FULLLSB*(MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA3*MBA2)
MBA5.T := BINMODE*ENCNT*/FULLLSB*(MBA4*MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA4*MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA4*MBA3*MBA2)
MBA6.T := BINMODE*ENCNT*/FULLLSB*(MBA5*MBA4*MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA5*MBA4*MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA5*MBA4*MBA3*MBA2)
MBA7.T := BINMODE*ENCNT*/FULLLSB*(MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA6*MBA5*MBA4*MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA6*MBA5*MBA4*MBA3*MBA2)
MBA8.T := BINMODE*ENCNT*/FULLLSB*(MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA7*MBA6*MBA5*MBA4*MBA3*MBA2)
MBA9.T :=
  BINMODE*ENCNT*/FULLLSB*(MBA8*MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*MBA0)+
  MODE512*ENCNT*/FULLLSB*(MBA8*MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1)+
  MODE1024*ENCNT*/FULLLSB*(MBA8*MBA7*MBA6*MBA5*MBA4*MBA3*MBA2)

LSEN = (MODE1X1+MODE2X2*MBA8+MODE4X4*MBA7*MBA8*MBA9)

MBA10.T := LS*ENCNT*/FULLMSB*LSEN
MBA11.T := LS*ENCNT*/FULLMSB*(MBA10*BINMODE*LSEN + MODE512)
MBA12.T := LS*ENCNT*/FULLMSB*
 (MBA10*MBA11*BINMODE*LSEN + MBA11*MODE512 + MODE1024)
MBA13.T := LS*BINMODE*LSEN*ENCNT*/FULLMSB*(MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA12)
MBA14.T := LS*BINMODE*LSEN*ENCNT*/FULLMSB*(MBA13*MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA13*MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA13*MBA12)
MBA15.T := LS*BINMODE*LSEN*ENCNT*/FULLMSB*(MBA14*MBA13*MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA14*MBA13*MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA14*MBA13*MBA12)
MBA16.T :=
  LS*BINMODE*LSEN*ENCNT*/FULLMSB*(MBA15*MBA14*MBA13*MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA15*MBA14*MBA13*MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA15*MBA14*MBA13*MBA12)
MBA17.T :=
  LS*BINMODE*LSEN*ENCNT*/FULLMSB*
  (MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA16*MBA15*MBA14*MBA13*MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA16*MBA15*MBA14*MBA13*MBA12)
MBA18.T :=
  LS*BINMODE*LSEN*ENCNT*/FULLMSB*
 (MBA17*MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*MBA10)+
  LS*MODE512*ENCNT*/FULLMSB*(MBA17*MBA16*MBA15*MBA14*MBA13*MBA12*MBA11)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA17*MBA16*MBA15*MBA14*MBA13*MBA12)
MBA19.T :=
  LS*BINMODE*LSEN*ENCNT*/FULLMSB*
   (MBA18*MBA17*MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*MBA10)+
  LS*MODE1024*ENCNT*/FULLMSB*(MBA18*MBA17*MBA16*MBA15*MBA14*MBA13*MBA12)

BRESLSB.CLKF = PIXCLK
BRESMSB.CLKF = PIXCLK

MBA0.CLKF = PIXCLK
MBA1.CLKF = PIXCLK
MBA2.CLKF = PIXCLK
MBA3.CLKF = PIXCLK
MBA4.CLKF = PIXCLK
MBA5.CLKF = PIXCLK
MBA6.CLKF = PIXCLK
MBA7.CLKF = PIXCLK
MBA8.CLKF = PIXCLK
MBA9.CLKF = PIXCLK
MBA10.CLKF = PIXCLK
MBA11.CLKF = PIXCLK
MBA12.CLKF = PIXCLK
MBA13.CLKF = PIXCLK
MBA14.CLKF = PIXCLK
MBA15.CLKF = PIXCLK
MBA16.CLKF = PIXCLK
MBA17.CLKF = PIXCLK
MBA18.CLKF = PIXCLK
MBA19.CLKF = PIXCLK

MM4A0.CLKF = PIXCLK
MM4A1.CLKF = PIXCLK
MM4A10.CLKF = PIXCLK
MM4A11.CLKF = PIXCLK

MM2A0.CLKF = PIXCLK
MM2A1.CLKF = PIXCLK
MM2A10.CLKF = PIXCLK
MM2A11.CLKF = PIXCLK

BRESLSB.TRST = VCC
BRESMSB.TRST = VCC

MBA0.TRST = BINMODE
MBA1.TRST = BINA512
MBA2.TRST = VCC
MBA3.TRST = VCC
MBA4.TRST = VCC
MBA5.TRST = VCC
MBA6.TRST = VCC
MBA7.TRST = VCC
MBA8.TRST = VCC
MBA9.TRST = VCC
MBA10.TRST = BINMODE
MBA11.TRST = BINA512
MBA12.TRST = VCC
MBA13.TRST = VCC
MBA14.TRST = VCC
MBA15.TRST = VCC
MBA16.TRST = VCC
MBA17.TRST = VCC
MBA18.TRST = VCC
MBA19.TRST = VCC


MBA0.RSTF = BRESLSB
MBA1.RSTF = BRESLSB
MBA2.RSTF = BRESLSB
MBA3.RSTF = BRESLSB
MBA4.RSTF = BRESLSB
MBA5.RSTF = BRESLSB
MBA6.RSTF = BRESLSB
MBA7.RSTF = BRESLSB
MBA8.RSTF = BRESLSB
MBA9.RSTF = BRESLSB
MBA10.RSTF = BRESMSB
MBA11.RSTF = BRESMSB
MBA12.RSTF = BRESMSB
MBA13.RSTF = BRESMSB
MBA14.RSTF = BRESMSB
MBA15.RSTF = BRESMSB
MBA16.RSTF = BRESMSB
MBA17.RSTF = BRESMSB
MBA18.RSTF = BRESMSB
MBA19.RSTF = BRESMSB

MM4A0.TRST = MODE1024
MM4A1.TRST = MODE1024
MM4A10.TRST = MODE1024
MM4A11.TRST = MODE1024

MM2A0.TRST = MODE512
MM2A1.TRST = GND
MM2A10.TRST = MODE512
MM2A11.TRST = GND

MM2A1.D := /LS*/FULLLSB*/MM2A1 + (LS+/ENCNT+FULLLSB)*MM2A1*/BRESLSB
MM2A11.D := LS*/FULLMSB*/MM2A11 + (/LS+/ENCNT+FULLMSB)*MM2A11*/BRESMSB


;CLKOUT = PIXCLK2 ; must be defined only for one FPGA
LSOUT.D := LS*(MODE1X1 + MODE2X2*MBA8 + MODE4X4*MBA9*MBA8*MBA7)
FSOUT.D := MS
LSOUT.CLKF = PIXCLK
FSOUT.CLKF = PIXCLK
```

```
LSOUT.TRST = M1R
FFSOUT.TRST = M1

SIMULATION

VECTOR COUNT := [MBA8, MBA7, MBA6, MBA5, MBA4, MBA3, MBA2, MBA1, MBA0]

TRACE_ON BRESLSB BRESMSB BKS0 BKS1 BKS2 BKS3 BINMODE FULLMSB FULLLSB
  CLKOUT FSOUT LSOUT MWE MOE MWR PIXCLK
  MBA8 MBA7 MBA6 MBA5 MBA4 MBA3 MBA2 MBA1 MBA0

SETF ENCNT /MS /FS /LS M0 /M1 PIXCLK /PIXCLK2

PRLDF /BRESLSB /BRESMSB /MBA9 MBA8 MBA7 /MBA6 /MBA5 /MBA4 /MBA3 /MBA2
PRLDF /MBA1 /MBA0 /MBA19 /MBA18 /MBA17 /MBA16 /MBA15 /MBA14 /MBA13 /MBA12
/MBA11 /MBA10

FOR X:=0 TO 127 DO
 BEGIN
  CLOCKF PIXCLK
 END
SETF LS
CLOCKF PIXCLK
SETF /LS
FOR X:=0 TO 10 DO
 BEGIN
  CLOCKF PIXCLK
 END

TRACE_OFF
```

```
TITLE video buffer
PATTERN
REVISION A
AUTHOR Robert Plante
COMPANY CRDV
DATE 29/11/94

CHIP VIDU1 iFX780_132

; Project Infra-red Eye
;
;
; This is the source file for one of the iFX780 on the Video Buffer (U1)
; The video buffer takes n (n=9) consecutive micro-scannned images
; of 256x256 pixels, store them in a proper sequence into a dual-buffer.
; When filled, the buffer is swapped with the other one to be fill.
; The first buffer is then scanned in a linear fashion to get a full
; oversampled image.

; This FPGA is used to generated the address and timings for the first
; buffer. This version work only for the 3x3 microscanning.


; version 1: first operational version
;---------------------------------- PIN Declarations --------------


PIN 118 /PIXCLK TTL_LEVEL

;Memory Buffer counters for address
PIN IO00 MBA0
PIN IO01 MBA1
PIN IO02 MBA2
PIN IO03 MBA3
PIN IO04 MBA4
PIN IO05 MBA5
PIN IO06 MBA6
PIN IO07 MBA7
PIN IO08 MBA8
PIN IO09 MBA9
PIN IO10 MBA10
PIN IO11 MBA11
PIN IO12 MBA12
PIN IO13 MBA13
PIN IO14 MBA14
PIN IO15 MBA15
PIN IO16 MBA16
PIN IO17 MBA17
PIN IO18 MBA18
PIN IO19 MBA19

;Memory Buffer counters for address, write mode 2x2, 4x4

PIN IO20 MM4A0     ; this output is used instead of MBA0
PIN IO21 MM4A1     ; this output is used instead of MBA1
PIN IO22 MM4A10    ; this output is used instead of MBA10
PIN IO23 MM2A11    ; this output is used instead of MBA11
PIN IO24 MM2A0     ; this output is used instead of MBA0
PIN IO25 MM2A1     ; not used
PIN IO26 MM2A10    ; this output is used instead of MBA0
PIN IO27 MM4A11    ; not used

PIN IO30 LSEN ; Intermediate value which enable counting LS pulses
PIN IO31 MWET ; Intermediate value to create a delay for MWE

; Spare pins, no connect -> program as output
PIN IO32 NC1

; Frame sync counter for generating MS sync
PIN IO33 FSCT0
PIN IO34 FSCT1
PIN IO35 FSCT2
PIN IO36 FSCT3
PIN IO37 /MSINT    ; MicroScan output
PIN IO38 MSTRG     ; External MS, must be AND with FS
PIN IO39 /MSINTEN  ; Enable internal MS generation

; Spare pins, no connect -> program as output
PIN IO50 NC2

PIN IO51 BRESMSB   ; Reset binary counter MSB
PIN IO52 /BKS0     ; BANK SELECT 0
PIN IO53 /BKS1     ; BANK SELECT 1
PIN IO54 /BKS2     ; BANK SELECT 2
PIN IO55 /BKS3     ; BANK SELECT 3
PIN IO56 BINMODE   ; Binary mode
PIN IO57 FULLMSB   ; Memory msb counter full

PIN IO70 FULLLSB   ; Memory lsb counter full
PIN IO71 MOR       ; Memory in read mode
PIN IO72 CLKOUT    ; Pixel clock out
PIN IO73 /FSOUT    ; Frame sync out
PIN IO74 /LSOUT    ; Line sync out
PIN IO75 /MWE      ; Memory buffer WE/
PIN IO76 /MOE      ; Memory buffer OE/
PIN IO77 /MWR      ; Memory transceiver enable for write
PIN IO78 ENCNT     ; Enable counting,may be an internal signal
PIN IO79 BRESLSB   ; Reset binary counter LSB


PIN IN0  /MS       ; Micro scanning sync pulse
PIN IN1  /FS       ; Frame sync pulse
PIN IN2  /LS       ; Line sync pulse
PIN IN3  M0        ; Mode bit0
PIN IN4  M1        ; Mode bit1
PIN IN5  PIXCLK2 TTL_LEVEL


STRING MOW ' (/MOR) '
STRING MODE1X1 ' (/M1 * /M0) '
STRING MODE2X2 ' (/M1 * M0) '
STRING MODE4X4 ' ( M1 * M0) '
STRING MODE512 ' (MODE2X2 * MOW) '
STRING MODE1024 ' (MODE4X4 * MOW) '

STATE
MOORE_MACHINE
DEFAULT_BRANCH HOLD_STATE

Z0 = /FSCT3 * /FSCT2 * /FSCT1 * /FSCT0
Z1 = /FSCT3 * /FSCT2 * /FSCT1 * FSCT0
Z2 = /FSCT3 * /FSCT2 * FSCT1 * /FSCT0
Z3 = /FSCT3 * /FSCT2 * FSCT1 * FSCT0
Z4 = /FSCT3 * FSCT2 * /FSCT1 * /FSCT0
Z5 = /FSCT3 * FSCT2 * /FSCT1 * FSCT0
Z6 = /FSCT3 * FSCT2 * FSCT1 * /FSCT0
Z7 = /FSCT3 * FSCT2 * FSCT1 * FSCT0
Z8 = FSCT3 * /FSCT2 * /FSCT1 * /FSCT0


Z0 := FS -> Z1
Z1 := FS -> Z2
Z2 := FS -> Z3
Z3 := FS -> Z4
Z4 := FS -> Z5
Z5 := FS -> Z6
Z6 := FS -> Z7
Z7 := FS -> Z8
Z8 := FS -> Z0


STATE
MOORE_MACHINE
DEFAULT_BRANCH HOLD_STATE

S0 = /MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
S1 = /MM4A11 * /MM4A10 * /MM4A1 * MM4A0
S2 = /MM4A11 * /MM4A10 * MM4A1 * /MM4A0
S3 = /MM4A11 * MM4A10 * /MM4A1 * /MM4A0
S4 = /MM4A11 * MM4A10 * /MM4A1 * MM4A0
S5 = /MM4A11 * MM4A10 * MM4A1 * /MM4A0
S6 = MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
S7 = MM4A11 * /MM4A10 * /MM4A1 * MM4A0
S8 = MM4A11 * /MM4A10 * MM4A1 * /MM4A0


S0 := FS*/MS -> S1
 + MS -> S0
S1 := FS*/MS -> S2
 + MS -> S0
S2 := FS*/MS -> S3
 + MS -> S0
S3 := FS*/MS -> S4
 + MS -> S0
S4 := FS*/MS -> S5
 + MS -> S0
S5 := FS*/MS -> S6
 + MS -> S0
S6 := FS*/MS -> S7
 + MS -> S0
S7 := FS*/MS -> S8
 + MS -> S0
S8 := MS -> S0


STATE
MOORE_MACHINE
DEFAULT_BRANCH HOLD_STATE

X0 = /MM2A10 * /MM2A0
X1 = /MM2A10 * MM2A0
X2 = MM2A10 * MM2A0
X3 = MM2A10 * /MM2A0

X0 := FS*/MS -> X1
 + MS -> X0
X1 := FS*/MS -> X2
 + MS -> X0
X2 := FS*/MS -> X3
 + MS -> X0
X3 := MS -> X0


EQUATIONS

NC1 = GND
NC2 = GND
MOR.D := /MOR*MS + MOR*/MS
MOE = MOR
MWR = /MOR
MWET = /MOR * /LS * PIXCLK2 * /(FULLLSB * FULLMSB)
MWE = MWET
MSINT = MSINTEN * FS *(FSCT3*/FSCT2*/FSCT1*/FSCT0) +
  /MSINTEN * (MSTRG * FS)
```

```
BINMODE = (MOR)
BRESLSB.D := MOR*LS*(MOR * MBA7*MBA8*MBA9) + MOW*LS
BRESMSB.D := (MOR*MS + MOW*FS)
FULLLSB = MOR * (/MBA0*MBA1*MBA2*MBA3*MBA4*MBA5*MBA6*MBA7*MBA8*MBA9) +
  MOW * (MBA2*MBA3*MBA4*MBA5*MBA6*MBA7*MBA8*MBA9)
FULLMSB = MOR*
  (/MBA10*MBA11*MBA12*MBA13*MBA14*MBA15*MBA16*MBA17*MBA18*MBA19) +
  MOW * (MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18* MBA19)

BKS0 = /MBA19 * /MBA18
BKS1 = /MBA19 * MBA18
BKS2 = MBA19 * /MBA18
BKS3 = MBA19 * MBA18

MBA0.D := ENCNT*/FULLLSB*/MBA0*/MBA1 + MBA0*(/ENCNT+FULLLSB)
MBA1.T := MOR*ENCNT*/FULLLSB*(MBA0 + MBA1)
MBA2.T := MOR*ENCNT*/FULLLSB*(MBA1*/MBA0)+
  MOW*ENCNT*/FULLLSB
MBA3.T := MOR*ENCNT*/FULLLSB*(MBA2*MBA1*/MBA0)+
  MOW*ENCNT*/FULLLSB*(MBA2)
MBA4.T := MOR*ENCNT*/FULLLSB*(MBA3*MBA2*MBA1*/MBA0)+
  MOW*ENCNT*/FULLLSB*(MBA3*MBA2)
MBA5.T := MOR*ENCNT*/FULLLSB*(MBA4*MBA3*MBA2*MBA1*/MBA0)+
  MOW*ENCNT*/FULLLSB*(MBA4*MBA3*MBA2)
MBA6.T := MOR*ENCNT*/FULLLSB*(MBA5*MBA4*MBA3*MBA2*MBA1*/MBA0)+
  MOW*ENCNT*/FULLLSB*(MBA5*MBA4*MBA3*MBA2)
MBA7.T := MOR*ENCNT*/FULLLSB*(MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*/MBA0)+
  MOW*ENCNT*/FULLLSB*(MBA6*MBA5*MBA4*MBA3*MBA2)
MBA8.T := MOR*ENCNT*/FULLLSB*(MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*/MBA0)+
  MOW*ENCNT*/FULLLSB*(MBA7*MBA6*MBA5*MBA4*MBA3*MBA2)
MBA9.T :=
  MOR*ENCNT*/FULLLSB*(MBA8*MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*/MBA0)+
  MOW*ENCNT*/FULLLSB*(MBA8*MBA7*MBA6*MBA5*MBA4*MBA3*MBA2)

LSEN = (MOR*MBA7*MBA8*MBA9)

MBA10.T := LS*ENCNT*/FULLMSB*LSEN*/MBA11
MBA11.T := LS*ENCNT*/FULLMSB*LSEN*(MBA10+MBA11)
MBA12.T := LS*ENCNT*/FULLMSB*(/MBA10*MBA11*LSEN + MOW)
MBA13.T := LS*LSEN*ENCNT*/FULLMSB*(MBA12*MBA11*/MBA10)+
  LS*MOW*ENCNT*/FULLMSB*(MBA12)
MBA14.T := LS*LSEN*ENCNT*/FULLMSB*(MBA13*MBA12*MBA11*/MBA10)+
  LS*MOW*ENCNT*/FULLMSB*(MBA13*MBA12)
MBA15.T := LS*LSEN*ENCNT*/FULLMSB*(MBA14*MBA13*MBA12*MBA11*/MBA10)+
  LS*MOW*ENCNT*/FULLMSB*(MBA14*MBA13*MBA12)
MBA16.T := LS*LSEN*ENCNT*/FULLMSB*(MBA15*MBA14*MBA13*MBA12*MBA11*/MBA10)+
  LS*MOW*ENCNT*/FULLMSB*(MBA15*MBA14*MBA13*MBA12)
MBA17.T :=
  LS*LSEN*ENCNT*/FULLMSB*(MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*/MBA10)+
  LS*MOW*ENCNT*/FULLMSB*(MBA16*MBA15*MBA14*MBA13*MBA12)
MBA18.T :=
  LS*LSEN*ENCNT*/FULLMSB*
  (MBA17*MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*/MBA10)+
  LS*MOW*ENCNT*/FULLMSB*(MBA17*MBA16*MBA15*MBA14*MBA13*MBA12)
MBA19.T :=
  LS*LSEN*ENCNT*/FULLMSB*
  (MBA18*MBA17*MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*/MBA10)+
  LS*MOW*ENCNT*/FULLMSB*(MBA18*MBA17*MBA16*MBA15*MBA14*MBA13*MBA12)


BRESLSB.CLKF = PIXCLK
BRESMSB.CLKF = PIXCLK
MOR.CLKF = PIXCLK

MBA0.CLKF = PIXCLK
MBA1.CLKF = PIXCLK
MBA2.CLKF = PIXCLK
MBA3.CLKF = PIXCLK
MBA4.CLKF = PIXCLK
MBA5.CLKF = PIXCLK
MBA6.CLKF = PIXCLK
MBA7.CLKF = PIXCLK
MBA8.CLKF = PIXCLK
MBA9.CLKF = PIXCLK
MBA10.CLKF = PIXCLK
MBA11.CLKF = PIXCLK
MBA12.CLKF = PIXCLK
MBA13.CLKF = PIXCLK
MBA14.CLKF = PIXCLK
MBA15.CLKF = PIXCLK
MBA16.CLKF = PIXCLK
MBA17.CLKF = PIXCLK
MBA18.CLKF = PIXCLK
MBA19.CLKF = PIXCLK

MM4A0.CLKF = PIXCLK
MM4A1.CLKF = PIXCLK
MM4A10.CLKF = PIXCLK
MM4A11.CLKF = PIXCLK

MM2A0.CLKF = PIXCLK
MM2A1.CLKF = PIXCLK
MM2A10.CLKF = PIXCLK
MM2A11.CLKF = PIXCLK

FSCT0.CLKF = /PIXCLK
FSCT1.CLKF = /PIXCLK
FSCT2.CLKF = /PIXCLK
FSCT3.CLKF = /PIXCLK

FSCT0.TRST = VCC
FSCT1.TRST = VCC
FSCT2.TRST = VCC
FSCT3.TRST = VCC

BRESLSB.TRST = VCC
BRESMSB.TRST = VCC
MOR.TRST = VCC
MSINT.TRST = VCC

MBA0.TRST = MOR
MBA1.TRST = MOR
MBA2.TRST = VCC
MBA3.TRST = VCC
MBA4.TRST = VCC
MBA5.TRST = VCC
MBA6.TRST = VCC
MBA7.TRST = VCC
MBA8.TRST = VCC
MBA9.TRST = VCC
MBA10.TRST = MOR
MBA11.TRST = MOR
MBA12.TRST = VCC
MBA13.TRST = VCC
MBA14.TRST = VCC
MBA15.TRST = VCC
MBA16.TRST = VCC
MBA17.TRST = VCC
MBA18.TRST = VCC
MBA19.TRST = VCC


MBA0.RSTF = BRESLSB
MBA1.RSTF = BRESLSB
MBA2.RSTF = BRESLSB
MBA3.RSTF = BRESLSB
MBA4.RSTF = BRESLSB
MBA5.RSTF = BRESLSB
MBA6.RSTF = BRESLSB
MBA7.RSTF = BRESLSB
MBA8.RSTF = BRESLSB
MBA9.RSTF = BRESLSB
MBA10.RSTF = BRESMSB
MBA11.RSTF = BRESMSB
MBA12.RSTF = BRESMSB
MBA13.RSTF = BRESMSB
MBA14.RSTF = BRESMSB
MBA15.RSTF = BRESMSB
MBA16.RSTF = BRESMSB
MBA17.RSTF = BRESMSB
MBA18.RSTF = BRESMSB
MBA19.RSTF = BRESMSB

MM4A0.TRST = MOW
MM4A1.TRST = MOW
MM4A10.TRST = MOW
MM4A11.TRST = MOW

MM2A0.TRST = GND
MM2A1.TRST = GND
MM2A10.TRST = GND
MM2A11.TRST = GND

MM2A1.D := GND
MM2A11.D := GND

CLKOUT = PIXCLK2
LSOUT.D := LS*(MOR*MBA9*MBA8*MBA7)
FSOUT.D := MS
LSOUT.CLKF = PIXCLK
FSOUT.CLKF = PIXCLK
LSOUT.TRST = MOR
FSOUT.TRST = MOR
```

```
TITLE video buffer
PATTERN
REVISION A
AUTHOR Robert Plante
COMPANY CRDV
DATE 9/11/94

CHIP VIDU1 iFX780_132

; Project Infra-red Eye
;
;
; This is the source file for one of the iFX780 on the Video Buffer (U11)
; The video buffer takes n (n=9) consecutive micro-scannned images
; of 256x256 pixels, store them in a proper sequence into a dual-buffer.
; When filled, the buffer is swapped with the other one to be fill.
; The first buffer is then scanned in a linear fashion to get a full
; oversampled image.

; This FPGA is used to generated the address and timings for the first
; buffer. This version work only for the 3x3 microscanning.

; version 1: first operational version
;--------------------------------- PIN Declarations --------------

PIN 118 /PIXCLK TTL_LEVEL

;Memory Buffer counters for address
PIN IO00 MBA0
PIN IO01 MBA1
PIN IO02 MBA2
PIN IO03 MBA3
PIN IO04 MBA4
PIN IO05 MBA5
PIN IO06 MBA6
PIN IO07 MBA7
PIN IO08 MBA8
PIN IO09 MBA9
PIN IO10 MBA10
PIN IO11 MBA11
PIN IO12 MBA12
PIN IO13 MBA13
PIN IO14 MBA14
PIN IO15 MBA15
PIN IO16 MBA16
PIN IO17 MBA17
PIN IO18 MBA18
PIN IO19 MBA19

;Memory Buffer counters for address, write mode 2x2, 4x4

PIN IO20 MM4A0    ; this output is used instead of MBA0
PIN IO21 MM4A1    ; this output is used instead of MBA1
PIN IO22 MM4A10   ; this output is used instead of MBA10
PIN IO23 MM2A11   ; this output is used instead of MBA11
PIN IO24 MM2A0    ; this output is used instead of MBA0
PIN IO25 MM2A1    ; not used
PIN IO26 MM2A10   ; this output is used instead of MBA0
PIN IO27 MM4A11   ; not used

PIN IO30 LSEN ; Intermediate value which enable counting LS pulses
PIN IO31 MWET ; Intermediate value to create a delay for MWE

; Spare pins, no connect -> program as output

PIN IO32 NC0
PIN IO33 NC1
PIN IO34 NC2
PIN IO35 NC3
PIN IO36 NC4
PIN IO37 NC5
PIN IO38 NC6
PIN IO39 NC7
PIN IO50 NC8

PIN IO51 BRESMSB  ; Reset binary counter MSB
PIN IO52 /BKS0    ; BANK SELECT 0
PIN IO53 /BKS1    ; BANK SELECT 1
PIN IO54 /BKS2    ; BANK SELECT 2
PIN IO55 /BKS3    ; BANK SELECT 3
PIN IO56 BINMODE  ; Binary mode
PIN IO57 FULLMSB  ; Memory msb counter full


PIN IO70 FULLLSB  ; Memory lsb counter full
PIN IO71 MOR      ; Memory in read mode
PIN IO72 CLKOUT   ; Pixel clock out
PIN IO73 /FSOUT   ; Frame sync out
PIN IO74 /LSOUT   ; Line sync out
PIN IO75 /MWE     ; Memory buffer WE/
PIN IO76 /MOE     ; Memory buffer OE/
PIN IO77 /MWR     ; Memory transceiver enable for write
PIN IO78 ENCNT    ; Enable counting,may be an internal signal
PIN IO79 BRESLSB  ; Reset binary counter LSB


PIN IN0  /MS      ; Micro scanning sync pulse
PIN IN1  /FS      ; Frame sync pulse
PIN IN2  /LS      ; Line sync pulse
PIN IN3  M0       ; Mode bit0
PIN IN4  M1       ; Mode bit1
PIN IN5  PIXCLK2 TTL_LEVEL

STRING M1R ' (/M0R) '
STRING M1W ' ( M0R) '

STATE
MOORE_MACHINE
DEFAULT_BRANCH HOLD_STATE

S0 = /MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
S1 = /MM4A11 * /MM4A10 * /MM4A1 * MM4A0
S2 = /MM4A11 * /MM4A10 * MM4A1 * /MM4A0
S3 = /MM4A11 * MM4A10 * /MM4A1 * /MM4A0
S4 = /MM4A11 * MM4A10 * /MM4A1 * MM4A0
S5 = /MM4A11 * MM4A10 * MM4A1 * /MM4A0
S6 = MM4A11 * /MM4A10 * /MM4A1 * /MM4A0
S7 = MM4A11 * /MM4A10 * /MM4A1 * MM4A0
S8 = MM4A11 * /MM4A10 * MM4A1 * /MM4A0


S0 := FS*/MS -> S1
 + MS -> S0
S1 := FS*/MS -> S2
 + MS -> S0
S2 := FS*/MS -> S3
 + MS -> S0
S3 := FS*/MS -> S4
 + MS -> S0
S4 := FS*/MS -> S5
 + MS -> S0
S5 := FS*/MS -> S6
 + MS -> S0
S6 := FS*/MS -> S7
 + MS -> S0
S7 := FS*/MS -> S8
 + MS -> S0
S8 := MS -> S0


STATE
MOORE_MACHINE
DEFAULT_BRANCH HOLD_STATE

X0 = /MM2A10 * /MM2A0
X1 = /MM2A10 * MM2A0
X2 = MM2A10 * MM2A0
X3 = MM2A10 * /MM2A0

X0 := FS*/MS -> X1
 + MS -> X0
X1 := FS*/MS -> X2
 + MS -> X0
X2 := FS*/MS -> X3
 + MS -> X0
X3 := MS -> X0


EQUATIONS

NC0 = GND
NC1 = GND
NC2 = GND
NC3 = GND
NC4 = GND
NC5 = GND
NC6 = GND
NC7 = GND
NC8 = GND

MOE = M1R
MWR = /M1R
MWET = /M1R * /LS * PIXCLK2 * /(FULLLSB * FULLMSB)
MWE = MWET
BINMODE = (M1R)
BRESLSB.D := M1R*LS*(M1R * MBA7*MBA8*MBA9) + M1W*LS
BRESMSB.D := (M1R*MS + M1W*FS)
FULLLSB = M1R * (/MBA0*MBA1*MBA2*MBA3*MBA4*MBA5*MBA6*MBA7*MBA8*MBA9) +
  M1W * (MBA2*MBA3*MBA4*MBA5*MBA6*MBA7*MBA8*MBA9)
FULLMSB = M1R *
  (/MBA10*MBA11*MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18* MBA19) +
  M1W * (MBA12*MBA13*MBA14*MBA15*MBA16*MBA17* MBA18* MBA19)
BKS0 = /MBA19 * /MBA18
BKS1 = /MBA19 * MBA18
BKS2 = MBA19 * /MBA18
BKS3 = MBA19 * MBA18

MBA0.D := ENCNT*/FULLLSB*/MBA0*/MBA1 + MBA0*(/ENCNT+FULLLSB)
MBA1.T := M1R*ENCNT*/FULLLSB*(MBA0 + MBA1)
MBA2.T := M1R*ENCNT*/FULLLSB*(MBA1*/MBA0)+
  M1W*ENCNT*/FULLLSB
MBA3.T := M1R*ENCNT*/FULLLSB*(MBA2*MBA1*/MBA0)+
  M1W*ENCNT*/FULLLSB*(MBA2)
MBA4.T := M1R*ENCNT*/FULLLSB*(MBA3*MBA2*MBA1*/MBA0)+
  M1W*ENCNT*/FULLLSB*(MBA3*MBA2)
MBA5.T := M1R*ENCNT*/FULLLSB*(MBA4*MBA3*MBA2*MBA1*/MBA0)+
  M1W*ENCNT*/FULLLSB*(MBA4*MBA3*MBA2)
MBA6.T := M1R*ENCNT*/FULLLSB*(MBA5*MBA4*MBA3*MBA2*MBA1*/MBA0)+
  M1W*ENCNT*/FULLLSB*(MBA5*MBA4*MBA3*MBA2)
MBA7.T := M1R*ENCNT*/FULLLSB*(MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*/MBA0)+
  M1W*ENCNT*/FULLLSB*(MBA6*MBA5*MBA4*MBA3*MBA2)
```

```
MBA8.T := M1R*ENCNT*/FULLLSB*(MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*/MBA0)+
  M1W*ENCNT*/FULLLSB*(MBA7*MBA6*MBA5*MBA4*MBA3*MBA2)
MBA9.T :=
  M1R*ENCNT*/FULLLSB*(MBA8*MBA7*MBA6*MBA5*MBA4*MBA3*MBA2*MBA1*/MBA0)+
  M1W*ENCNT*/FULLLSB*(MBA8*MBA7*MBA6*MBA5*MBA4*MBA3*MBA2)

LSEN = (M1R*MBA7*MBA8*MBA9)
MBA10.T := LS*ENCNT*/FULLMSB*LSEN*/MBA11
MBA11.T := LS*ENCNT*/FULLMSB*LSEN*(MBA10+MBA11)
MBA12.T := LS*ENCNT*/FULLMSB*(/MBA10*MBA11*LSEN + M1W)
MBA13.T := LS*LSEN*ENCNT*/FULLMSB*(MBA12*MBA11*/MBA10)+
  LS*M1W*ENCNT*/FULLMSB*(MBA12)
MBA14.T := LS*LSEN*ENCNT*/FULLMSB*(MBA13*MBA12*MBA11*/MBA10)+
  LS*M1W*ENCNT*/FULLMSB*(MBA13*MBA12)
MBA15.T := LS*LSEN*ENCNT*/FULLMSB*(MBA14*MBA13*MBA12*MBA11*/MBA10)+
  LS*M1W*ENCNT*/FULLMSB*(MBA14*MBA13*MBA12)
MBA16.T := LS*LSEN*ENCNT*/FULLMSB*(MBA15*MBA14*MBA13*MBA12*MBA11*/MBA10)+
  LS*M1W*ENCNT*/FULLMSB*(MBA15*MBA14*MBA13*MBA12)
MBA17.T :=
  LS*LSEN*ENCNT*/FULLMSB*(MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*/MBA10)+
  LS*M1W*ENCNT*/FULLMSB*(MBA16*MBA15*MBA14*MBA13*MBA12)
MBA18.T :=
  LS*LSEN*ENCNT*/FULLMSB*
  (MBA17*MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*/MBA10)+
  LS*M1W*ENCNT*/FULLMSB*(MBA17*MBA16*MBA15*MBA14*MBA13*MBA12)
MBA19.T :=
  LS*LSEN*ENCNT*/FULLMSB*
  (MBA18*MBA17*MBA16*MBA15*MBA14*MBA13*MBA12*MBA11*/MBA10)+
  LS*M1W*ENCNT*/FULLMSB*(MBA18*MBA17*MBA16*MBA15*MBA14*MBA13*MBA12)

BRESLSB.CLKF = PIXCLK
BRESMSB.CLKF = PIXCLK

MBA0.CLKF = PIXCLK
MBA1.CLKF = PIXCLK
MBA2.CLKF = PIXCLK
MBA3.CLKF = PIXCLK
MBA4.CLKF = PIXCLK
MBA5.CLKF = PIXCLK
MBA6.CLKF = PIXCLK
MBA7.CLKF = PIXCLK
MBA8.CLKF = PIXCLK
MBA9.CLKF = PIXCLK
MBA10.CLKF = PIXCLK
MBA11.CLKF = PIXCLK
MBA12.CLKF = PIXCLK
MBA13.CLKF = PIXCLK
MBA14.CLKF = PIXCLK
MBA15.CLKF = PIXCLK
MBA16.CLKF = PIXCLK
MBA17.CLKF = PIXCLK
MBA18.CLKF = PIXCLK
MBA19.CLKF = PIXCLK

MM4A0.CLKF = PIXCLK
MM4A1.CLKF = PIXCLK
MM4A10.CLKF = PIXCLK
MM4A11.CLKF = PIXCLK

MM2A0.CLKF = PIXCLK
MM2A1.CLKF = PIXCLK
MM2A10.CLKF = PIXCLK
MM2A11.CLKF = PIXCLK

BRESLSB.TRST = VCC
BRESMSB.TRST = VCC

MBA0.TRST = M1R
MBA1.TRST = M1R
MBA2.TRST = VCC
MBA3.TRST = VCC
MBA4.TRST = VCC
MBA5.TRST = VCC
MBA6.TRST = VCC
MBA7.TRST = VCC
MBA8.TRST = VCC
MBA9.TRST = VCC
MBA10.TRST = M1R
MBA11.TRST = M1R
MBA12.TRST = VCC
MBA13.TRST = VCC
MBA14.TRST = VCC
MBA15.TRST = VCC
MBA16.TRST = VCC
MBA17.TRST = VCC
MBA18.TRST = VCC
MBA19.TRST = VCC


MBA0.RSTF = BRESLSB
MBA1.RSTF = BRESLSB
MBA2.RSTF = BRESLSB
MBA3.RSTF = BRESLSB
MBA4.RSTF = BRESLSB
MBA5.RSTF = BRESLSB
MBA6.RSTF = BRESLSB
MBA7.RSTF = BRESLSB
MBA8.RSTF = BRESLSB
MBA9.RSTF = BRESLSB
MBA10.RSTF = BRESMSB
MBA11.RSTF = BRESMSB
MBA12.RSTF = BRESMSB
MBA13.RSTF = BRESMSB
MBA14.RSTF = BRESMSB
MBA15.RSTF = BRESMSB
MBA16.RSTF = BRESMSB
MBA17.RSTF = BRESMSB
MBA18.RSTF = BRESMSB
MBA19.RSTF = BRESMSB

MM4A0.TRST = M1W
MM4A1.TRST = M1W
MM4A10.TRST = M1W
MM4A11.TRST = M1W

MM2A0.TRST = GND
MM2A1.TRST = GND
MM2A10.TRST = GND
MM2A11.TRST = GND

MM2A1.D := GND
MM2A11.D := GND

CLKOUT = PIXCLK2
LSOUT.D := LS*(M1R*MBA9*MBA8*MBA7)
FSOUT.D := MS
LSOUT.CLKF = PIXCLK
FSOUT.CLKF = PIXCLK
LSOUT.TRST = M1R
FSOUT.TRST = M1R
```

APPENDIX E

DIPIX Configuration File

```
;================================================
; Module : 256x256.cpf
;
; Purpose : cpf for Pro-View in microscanning mode 1
;================================================
[XPG]
CameraName =Amber
ComPortCount =1
BytesPerPort =2
SDIBSW0 =0x00
SDIBSW1 =0x00
SDIBSW2 =0x00
ICPLFile1 =C11620.JED
ICPLFile2 =C11727.JED
ICPLFile3 =D11728.JED
ModuleType =P_MODULE_TYPE_DCM

[FRAME TIMING]
PixelsPerClock =1
BitsPerPixel =16
LineSize =256
FieldSize =268
HVideoStart =0
HVideoActive =256
VVideoStart =0
VVideoActive =256
HAcquireStart =4
HAcquireActive =252
VAcquireStart =0
VAcquireActive =256

[CAMERA CONTROL]
MultiImageGrab =P_SINGLE_IMAGE_GRAB
FrameCount =1
CameraType =P_CAMERATYPE_AREASCAN
ImageFormat =P_IMAGE_FORMAT_RASTER
NumFields =1

[CAMERA MODULE]
SDIBBusID =0
FrameValidSync =P_SYNC_SYNCHRONOUS
FrameValidPolarity =P_POLARITY_ACTIVE_LOW
LineValidSync =P_SYNC_SYNCHRONOUS
LineValidPolarity =P_POLARITY_ACTIVE_LOW
PixelSignalFormat =P_SIGNAL_FORMAT_RS422
CameraNumber =1
ClockSelect =P_CLOCKSRC_0
Clock0Format =P_SIGNAL_FORMAT_RS422
ClockPolarity =P_POLARITY_FALLING_EDGE
ClockDelay =P_CLOCK_DELAY_NONE
OscillatorType =P_OSCILLATOR_NONE
IOFormatIn01 =P_SIGNAL_FORMAT_RS422
IOFormatIn23 =P_SIGNAL_FORMAT_RS422
IOFormatIn45 =P_SIGNAL_FORMAT_RS422
IOFormatIn67 =P_SIGNAL_FORMAT_RS422
IOFormatOut01 =P_SIGNAL_FORMAT_RS422
IOFormatOut23 =P_SIGNAL_FORMAT_RS422
IOFormatOut45 =P_SIGNAL_FORMAT_RS422
IOFormatOut67 =P_SIGNAL_FORMAT_RS422

;================================================
; Module : 512x512.cpf
;
; Purpose : cpf for Pro-View in microscanning mode 2
;================================================
[XPG]
CameraName =Amber
ComPortCount =1
BytesPerPort =2
SDIBSW0 =0x00
SDIBSW1 =0x00
SDIBSW2 =0x00
ICPLFile1 =C11620.JED
ICPLFile2 =C11727.JED
ICPLFile3 =D11728.JED
ModuleType =P_MODULE_TYPE_DCM

[FRAME TIMING]
PixelsPerClock =1
BitsPerPixel =16
LineSize =512
FieldSize =536
HVideoStart =0
HVideoActive =512
VVideoStart =0
VVideoActive =512
HAcquireStart =8
HAcquireActive =498
VAcquireStart =0
VAcquireActive =512

[CAMERA CONTROL]
MultiImageGrab =P_SINGLE_IMAGE_GRAB
FrameCount =1
CameraType =P_CAMERATYPE_AREASCAN
ImageFormat =P_IMAGE_FORMAT_RASTER
NumFields =1

[CAMERA MODULE]
SDIBBusID =0
FrameValidSync =P_SYNC_SYNCHRONOUS
FrameValidPolarity =P_POLARITY_ACTIVE_LOW
LineValidSync =P_SYNC_SYNCHRONOUS
LineValidPolarity =P_POLARITY_ACTIVE_LOW
PixelSignalFormat =P_SIGNAL_FORMAT_RS422
CameraNumber =1
ClockSelect =P_CLOCKSRC_0
Clock0Format =P_SIGNAL_FORMAT_RS422
ClockPolarity =P_POLARITY_FALLING_EDGE
ClockDelay =P_CLOCK_DELAY_NONE
OscillatorType =P_OSCILLATOR_NONE
IOFormatIn01 =P_SIGNAL_FORMAT_RS422
IOFormatIn23 =P_SIGNAL_FORMAT_RS422
IOFormatIn45 =P_SIGNAL_FORMAT_RS422
IOFormatIn67 =P_SIGNAL_FORMAT_RS422
IOFormatOut01 =P_SIGNAL_FORMAT_RS422
IOFormatOut23 =P_SIGNAL_FORMAT_RS422
IOFormatOut45 =P_SIGNAL_FORMAT_RS422
IOFormatOut67 =P_SIGNAL_FORMAT_RS422

;================================================
; Module : 768x768.cpf
;
; Purpose : cpf for Pro-View in microscanning mode 3
;================================================
[XPG]
CameraName =Amber
ComPortCount =1
BytesPerPort =2
SDIBSW0 =0x00
SDIBSW1 =0x00
SDIBSW2 =0x00
ICPLFile1 =C11620.JED
ICPLFile2 =C11727.JED
ICPLFile3 =D11728.JED
ModuleType =P_MODULE_TYPE_DCM

[FRAME TIMING]
PixelsPerClock =1
BitsPerPixel =16
LineSize =768
FieldSize =804
HVideoStart =0
HVideoActive =768
VVideoStart =0
VVideoActive =768
HAcquireStart =12
HAcquireActive =756
VAcquireStart =0
VAcquireActive =768

[CAMERA CONTROL]
MultiImageGrab =P_SINGLE_IMAGE_GRAB
```

```
FrameCount =1
CameraType =P_CAMERATYPE_AREASCAN
ImageFormat =P_IMAGE_FORMAT_RASTER
NumFields =1

[CAMERA MODULE]
SDIBBusID =0
FrameValidSync =P_SYNC_SYNCHRONOUS
FrameValidPolarity =P_POLARITY_ACTIVE_LOW
LineValidSync =P_SYNC_SYNCHRONOUS
LineValidPolarity =P_POLARITY_ACTIVE_LOW
PixelSignalFormat =P_SIGNAL_FORMAT_RS422
CameraNumber =1
ClockSelect =P_CLOCKSRC_0
Clock0Format =P_SIGNAL_FORMAT_RS422
ClockPolarity =P_POLARITY_FALLING_EDGE
ClockDelay =P_CLOCK_DELAY_NONE
OscillatorType =P_OSCILLATOR_NONE
IOFormatIn01 =P_SIGNAL_FORMAT_RS422
IOFormatIn23 =P_SIGNAL_FORMAT_RS422
IOFormatIn45 =P_SIGNAL_FORMAT_RS422
IOFormatIn67 =P_SIGNAL_FORMAT_RS422
IOFormatOut01 =P_SIGNAL_FORMAT_RS422
IOFormatOut23 =P_SIGNAL_FORMAT_RS422
IOFormatOut45 =P_SIGNAL_FORMAT_RS422
IOFormatOut67 =P_SIGNAL_FORMAT_RS422

;================================================
; Module : 1024x1024.cpf
;
; Purpose : cpf for Pro-View in microscanning mode 4
;================================================
[XPG]
CameraName =Amber
ComPortCount =1
BytesPerPort =2
SDIBSW0 =0x00
SDIBSW1 =0x00
SDIBSW2 =0x00
ICPLFile1 =C11620.JED
ICPLFile2 =C11727.JED
ICPLFile3 =D11728.JED
ModuleType =P_MODULE_TYPE_DCM

[FRAME TIMING]
PixelsPerClock =1
BitsPerPixel =16
LineSize =1024
FieldSize =1072
HVideoStart =0
HVideoActive =1024
VVideoStart =0
VVideoActive =1024
HAcquireStart =16
HAcquireActive =1008
VAcquireStart =0
VAcquireActive =1024

[CAMERA CONTROL]
MultiImageGrab =P_SINGLE_IMAGE_GRAB
FrameCount =1
CameraType =P_CAMERATYPE_AREASCAN
ImageFormat =P_IMAGE_FORMAT_RASTER
NumFields =1

[CAMERA MODULE]
SDIBBusID =0
FrameValidSync =P_SYNC_SYNCHRONOUS
FrameValidPolarity =P_POLARITY_ACTIVE_LOW
LineValidSync =P_SYNC_SYNCHRONOUS
LineValidPolarity =P_POLARITY_ACTIVE_LOW
PixelSignalFormat =P_SIGNAL_FORMAT_RS422
CameraNumber =1
ClockSelect =P_CLOCKSRC_0
Clock0Format =P_SIGNAL_FORMAT_RS422
ClockPolarity =P_POLARITY_FALLING_EDGE
ClockDelay =P_CLOCK_DELAY_NONE
OscillatorType =P_OSCILLATOR_NONE
IOFormatIn01 =P_SIGNAL_FORMAT_RS422
IOFormatIn23 =P_SIGNAL_FORMAT_RS422
IOFormatIn45 =P_SIGNAL_FORMAT_RS422
IOFormatIn67 =P_SIGNAL_FORMAT_RS422
IOFormatOut01 =P_SIGNAL_FORMAT_RS422
IOFormatOut23 =P_SIGNAL_FORMAT_RS422
IOFormatOut45 =P_SIGNAL_FORMAT_RS422
IOFormatOut67 =P_SIGNAL_FORMAT_RS422
```

INTERNAL DISTRIBUTION

DREV - TM - 9813

1- Deputy Director General
1- Chief Scientist
1- Head/Passive Systems for Surveillance and Target Acquisition
1- Head/Active Systems for Surveillance and Target Acquisition
1- BDO (R. Larose)
6- Document Library
1- J. Fortin (author)
1- P. Chevrette (author)
1- D. St-Germain
1- J. Delisle
1- B. Montminy
1- V. Larochelle
1- J. Dubois
1- J.-P. Ardouin
1- J. Cruickshank
1- J. Maheux
1- J. Boulter
1- M. Lévesque
1- J. Dumas
1- A. Blanchard
1- L. Sévigny

EXTERNAL DISTRIBUTION

DREV - TM - 9813

1- DRDIM
1- DRDIM unbound copy
1- DRDB
1- SAA
1- DSAA
1- DSAA 3
1- DSAL
1- DSAM
1- DAR
1- DAR 2-3 (Maj T. Lindsey)
1- DAEPM (C) 4-8
2- DAEPM (C) 4-8-2
1- DAEPM (C) 4-8-6
1- DAEPM 5-5-5
5- NSS
1- TOTEF, CFB Trebton (Capt G. Ralf)
1- PFPTO 426 Squadron/TOTEF
1- PMO Maritime Helicopter Patrol (MWO G. Williams)

DOCUMENT CONTROL DATA

1. ORIGINATOR (name and address) Jean Fortin	2. SECURITY CLASSIFICATION (Including special warning terms if applicable) Unclassified

3. TITLE (Its classification should be indicated by the appropriate abbreviation (S, C, R or U)
INFRARED EYE: MICROSCANNING (U)

4. AUTHORS (Last name, first name, middle initial. If military, show rank, e.g. Doe, Maj. John E.)
Fortin Jean, Paul Chevrette

5. DATE OF PUBLICATION (month and year) 06/98	6a. NO. OF PAGES 70	6b. NO. OF REFERENCES 16

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. Give the inclusive dates when a specific reporting period is covered.)
Memorandum

8. SPONSORING ACTIVITY (name and address)
Defence Research Establishment, Valcartier
2459, Boul Pie XI North
Val Bélair, Qc, G3J 1X5

9a. PROJECT OR GRANT NO. (Please specify whether project or grant) Project: 3DA14 NSI FE: 847792FACDA	9b. CONTRACT NO. N/A
10a. ORIGINATOR'S DOCUMENT NUMBER DREV-TM-9813	10b. OTHER DOCUMENT NOS N/A

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

- [x] Unlimited distribution
- [ ] Contractors in approved countries (specify)
- [ ] Canadian contractors (with need-to-know)
- [ ] Government (with need-to-know)
- [ ] Defense departments
- [ ] ~~Other (please specify)~~

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

This memorandum describes the research and development work that has been undertaken to develop and build a microscanning camera system. The work was conducted as part of the Infrared Eye project which shows a new concept of an infrared surveillance camera system aimed at improving the effectiveness of search an rescue operations. The system is based on two IR cameras operating simultaneously, one covering a wide field-of-view (WFOV) and the other one having a narrow field-of-view (NFOV) mobile within the WFOV. The WFOV camera is optimized for detection while the NFOV camera is optimized for resolution. The resolution of that camera is improved using a technique based on microscanning.

The microscanning technique described in this document and implemented in the project differs from others currently used and is the object of a patent request. It consists in imposing micro-displacements to the imaging lens in front of the focal plane array (FPA) by means of piezo-electric micro-positioners and special flexible supports for the lens in both X and Y axis. This support method allows for a symmetrical mechanical loading for the two axis.

The advantages of the developed microscanning technique and device are that it can be synchronized to the adjustable frame rate and integration time of the camera, and any microscanning mode and path can be programmed electronically. In our application, we can select 2x2, 3x3 and 4x4 microscanning steps, as opposed to more conventional techniques using a wheel of prisms with fixed speed and mode. Also, the obtained image displacement is equal to the displacement imposed to the lens, irrelevant of its focal length, hence allowing a change of the lens or telescope which may be used as front optics without having to modify the microscanning system.

The implementation of this microscanning technique for the NFOV camera in the IR Eye project significantly improved its resolution and therefore its performance as a component of the system.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Microscanning, Focal plane array, Infrared, Infrared sensors, Infrared detectors, Resolution

UNCLASSIFIED

Requests for documents
should be sent to:

DIRECTORATE RESEARCH AND DEVELOPMENT
INFORMATION MANAGEMENT
Dept. of National Defence
Ottawa, Ontario
K1A 0K2

Tel.: (613) 995-2971
Fax: (613) 996-0392

#508740

Toute demande de document
doit être adressée à:

DIRECTEUR-GESTION DE L'INFORMATION DE RECHERCHE
ET DE DÉVELOPPEMENT
Ministère de la Défense nationale
Ottawa, Ontario
K1A 0K2

Téléphone: (613) 995-2971
Télécopieur: (613) 996-0392

SANS CLASSIFICATION